



Sybase, Inc.
One Sybase Drive
Dublin, CA 94568
www.sybase.com

Sybase Avaki EII Administration Guide

Release 7.0

August 24, 2006

This publication pertains to Sybase software and to any subsequent release until otherwise indicated in new editions or technical notes. Information in this document is subject to change without notice. The software described herein is furnished under a license agreement, and it may be used or copied only in accordance with the terms of that agreement.

To order additional documents, U.S. and Canadian customers should call Customer Fulfillment at (800) 685-8225, fax (617) 229-9845.

Customers in other countries with a U.S. license agreement may contact Customer Fulfillment via the above fax number. All other international customers should contact their Sybase subsidiary or local distributor. Upgrades are provided only at regularly scheduled software release dates. No part of this publication may be reproduced, transmitted, or translated in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without the prior written permission of Sybase, Inc.

Sybase, SYBASE (logo), ADA Workbench, Adaptable Windowing Environment, Adaptive Component Architecture, Adaptive Server, Adaptive Server Anywhere, Adaptive Server Enterprise, Adaptive Server Enterprise Monitor, Adaptive Server Enterprise Replication, Adaptive Server Everywhere, Advantage Database Server, Afaria, Answers Anywhere, Applied Meta, Applied Metacomputing, AppModeler, APT Workbench, APT-Build, APT-Edit, APT-Execute, APT-Translator, APT-Library, ASEP, Avaki, Avaki (Arrow Design), Avaki Data Grid, AvantGo, Backup Server, BayCam, Beyond Connected, Bit-Wise, BizTracker, Certified PowerBuilder Developer, Certified SYBASE Professional, Certified SYBASE Professional Logo, ClearConnect, Client-Library, Client Services, CodeBank, Column Design, ComponentPack, Connection Manager, Convoy/DM, Copernicus, CSP, Data Pipeline, Data Workbench, DataArchitect, Database Analyzer, DataExpress, DataServer, DataWindow, DataWindow .NET, DB-Library, dbQueue, Dejima, Dejima Direct, Developers Workbench, DirectConnect Anywhere, DirectConnect, Distribution Director, Dynamic Mobility Model, e-ADK, E-Anywhere, e-Biz Integrator, E-Whatever, EC Gateway, ECMap, ECRT, eFulfillment Accelerator, EII Plus, Electronic Case Management, Embedded SQL, EMS, Enterprise Application Studio, Enterprise Client/Server, Enterprise Connect, Enterprise Data Studio, Enterprise Manager, Enterprise Portal (logo), Enterprise SQL Server Manager, Enterprise Work Architecture, Enterprise Work Designer, Enterprise Work Modeler, eProcurement Accelerator, eremote, Everything Works Better When Everything Works Together, EWA, ExtendedAssist, Extended Systems, ExtendedView, Financial Fusion, Financial Fusion (and design), Financial Fusion Server, Formula One, Fusion Powered e-Finance, Fusion Powered Financial Destinations, Fusion Powered STP, Gateway Manager, GeoPoint, GlobalFIX, iAnywhere, iAnywhere Solutions, ImpactNow, Industry Warehouse Studio, InfoMaker, Information Anywhere, Information Everywhere, InformationConnect, InstaHelp, Intelligent Self-Care, InternetBuilder, iremote, iScript, Jaguar CTS, jConnect for JDBC, KnowledgeBase, Legion, Logical Memory Manager, lLite, M2M Anywhere, Mach Desktop, Mail Anywhere Studio, Mainframe Connect, Maintenance Express, Manage Anywhere Studio, MAP, M-Business Anywhere, M-Business Channel, M-Business Network, M-Business Suite, MDI Access Server, MDI Database Gateway, media.splash, Message Anywhere Server, MetaWorks, MethodSet, mFolio, Mirror Activator, ML Query, MobiCATS, MobileQ, MySupport, Net-Gateway, Net-Library, New Era of Networks, Next Generation Learning, Next Generation Learning Studio, O DEVICE, OASIS, OASIS logo, ObjectConnect, ObjectCycle, OmniConnect, OmniQ, OmniSQL Access Module, OmniSQL Toolkit, OneBridge, Open Biz, Open Business Interchange, Open Client, Open ClientConnect, Open Client/Server, Open Client/Server Interfaces, Open Gateway, Open Server, Open ServerConnect, Open Solutions, Optima++, Partnerships that Work, PB-Gen, PC APT Execute, PC DB-Net, PC Net Library, Pharma Anywhere, PhysicalArchitect, Pocket PowerBuilder, PocketBuilder, Power++, Power Through Knowledge, power.stop, PowerAMC, PowerBuilder, PowerBuilder Foundation Class Library, PowerDesigner, PowerDimensions, PowerDynamo, Powering the New Economy, PowerScript, PowerSite, PowerSocket, Powersoft, PowerStage, PowerStudio, PowerTips, Powersoft Portfolio, Powersoft Professional, PowerWare Desktop, PowerWare Enterprise, ProcessAnalyst, Pylon, Pylon Anywhere, Pylon Application Server, Pylon Conduit, Pylon PIM Server, Pylon Pro, QAnywhere, Rapport, Relational Beans, RemoteWare, RepConnector, Report Workbench, Report-Execute, Replication Agent, Replication Driver, Replication Server, Replication Server Manager, Replication Toolkit, Resource Manager, RFID Anywhere, RW-DisplayLib, RW-Library, SAFE, SAFE/PRO, Sales Anywhere, Search Anywhere, SDF, Search Anywhere, Secure SQL Server, Secure SQL Toolkit, Security Guardian, ShareLink, SharePool, SKILLS, smart.partners, smart.parts, smart.script, SOA Anywhere Trademark, SQL Advantage, SQL Anywhere, SQL Anywhere Studio, SQL Code Checker, SQL Debug, SQL Edit, SQL Edit/TPU, SQL Everywhere, SQL Modeler, SQL Remote, SQL Server, SQL Server Manager, SQL SMART, SQL Toolkit, SQL Server/CFT, SQL Server/DBM, SQL Server SNMP SubAgent, SQL Station, SQLJ, Stage III Engineering, Startup.Com, STEP, SupportNow, S.W.I.F.T. Message Format Libraries, Sybase Central, Sybase Client/Server Interfaces, Sybase Development Framework, Sybase Financial Server, Sybase Gateways, Sybase Learning Connection, Sybase MPP, Sybase SQL Desktop, Sybase SQL Lifecycle, Sybase SQL Workgroup, Sybase Synergy Program, Sybase Virtual Server Architecture, Sybase User Workbench, SybaseWare, Syber Financial, SyberAssist, SybFlex, SybMD, SyBooks, System 10, System 11, System XI (logo), SystemTools, Tabular Data Stream, The Enterprise Client/Server Company, The Extensible Software Platform, The Future Is Wide Open, The Learning Connection, The Model For Client/Server Solutions, The Online Information Center, The Power of One, TotalFix, TradeForce, Transact-SQL, Translation Toolkit, Turning Imagination Into Reality, UltraLite, UltraLite.NET, UNIBOM, Unilib, Uninull, Unisep, Unistring, URK Runtime Kit for UniCode, Viofone, Viewer, VisualWriter, VQL, WarehouseArchitect, Warehouse Control Center, Warehouse Studio, Warehouse WORKS, Watcom, Watcom SQL, Watcom SQL Server, Web Deployment Kit, Web.PB, Web.SQL, WebSights, WebViewer, WorkGroup SQL Server, XA-Library, XA-Server, XcelleNet, XP Server, XTNDAccess and XTNDConnect are trademarks of Sybase, Inc. or its subsidiaries. 07/06

Unicode and the Unicode Logo are registered trademarks of Unicode, Inc.

All other company and product names used herein may be trademarks or registered trademarks of their respective companies.

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

Set in *Arial*, *Courier New*, and *Times New Roman*. Stanley Morison, the creator of *Times New Roman*, said of it: “By the vice of Mammon and the misery of the machine, it is bigoted and narrow, mean and puritan.”

Credits

This product includes software developed by the Apache Software Foundation (<http://www.apache.org>). This product includes Hypersonic SQL and ANTLR. This product includes code licenses from RSA Security, Inc. Some portions licensed from IBM are available at <http://oss.software.ibm.com/icu4j/>. Contains IBM® 64-bit Runtime Environment for AIX™, Java™ 2 Technology Edition Version 1.4 Modules © Copyright IBM Corporation 1999, 2000 All Rights Reserved. Contains the SAXON XSLT Processor from Michael Kay, which is available at <http://saxon.sourceforge.net>. This product includes software developed by the Proxool Project (<http://proxool.sourceforge.net>).

Sybase Avaki EII Administration Guide

Written by Beth Thoenen, Cheryl Magadiou, and Anand Natrajan

Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

Table of contents

Preface xi

Audience **xi**

Organization **xii**

Related documentation and online help **xiii**

 Manuals **xiii**

 Online help **xiii**

Conventions **xv**

 Command syntax conventions **xv**

 Conventions for screen examples **xvi**

How to contact Avaki support at Sybase, Inc. **xvii**

Chapter 1

Planning your Avaki deployment 1

Planning steps **2**

System requirements **3**

 Databases and JDBC drivers **6**

Planning use of network ports **8**

Planning for SSL certificates **10**

Creating a DNS alias for the GDC machine **11**

Configuring user accounts **11**

 User accounts for enabling auto-restart **12**

 User accounts for servers supporting Avaki shares **12**

Chapter 2

Installing Avaki software 13

Overview of the installation process **14**

Loading and starting Avaki Data Grid software **15**

 Installing in Unix **16**

 Before installing on a system running any version of Unix **16**

 Installing Avaki software **16**

Installing in Windows	22
Preinstallation steps for Windows 2003	22
Windows installation steps	24
How to proceed	31
Setting a server's host name or IP address	32
Host name changes	33
Setting the host name property	33
Generating self-signed SSL certificates	34
Starting a primary GDC	37
Using nondefault ports	37
Auto-restart or not?	37
Starting the GDC without auto-restart	38
Starting the GDC with auto-restart	38
Installing an SSL certificate	39
Configuring a primary GDC	41
Starting a secondary GDC	43
Setting up administrative accounts in Avaki	45
Administrative groups	45
Overview: Setting up Avaki administrators	46
How to proceed	46

Chapter 3

<i>Setting up Avaki servers and clients</i>	47
Server and client requirements	48
Setting up grid servers	48
Installing Avaki software	48
Installing database JDBC drivers	49
Configuring a grid server to use nondefault ports	50
Starting a grid server	51
Starting the grid server without auto-restart	51
Starting the grid server with auto-restart	51
Configuring a grid server	52
Setting up share servers	54
About share servers	54
Benefits of share servers	54
Multiple share servers	55
Setting up a single share server	56
Configuring multiple share servers on one machine OR configuring share servers to use nondefault ports	59
Setting up data grid access servers	62
About data grid access servers	62

System requirements for data grid access servers	64
Overview of the data grid access server setup procedure	64
Preparing to run a data grid access server	65
On a machine that will serve NFS clients	65
On a machine that will serve CIFS clients	67
Setting up user accounts and groups	68
Setting domain-wide user and group mappings	68
Setting up user account mappings	70
Setting up group mappings	71
Setting up default mappings	71
Changing the host name	73
Starting the data grid access server	73
Configuring the DGAS to use nondefault ports	76
Creating a DGAS properties file	76
Using the DGAS properties file	78
Connecting the DGAS to your Avaki domain	79
Configuring DGAS properties	82
Restarting the DGAS	85
Setting admission policies	86
Admission policy rules	86
LDAP authentication	86
Configuring admission policies	87
Setting per-DGAS user and group mappings	88
Configuring a cache service for a DGAS	90
Setting up NFS clients	92
System requirements for NFS clients	92
NFS-mounting Avaki directories	92
Examples	93
Setting up CIFS clients	94
Setting up proxy servers and routing tables	94
Setting up command clients	95
Chapter 4	
<i>Managing Avaki servers</i>	97
Categories	98
Displaying domain information	98
Finding your version number	99
Monitoring	99
Viewing the operations status screen	99
Using monitor services	100

Setting up a monitor service	101
Setting up monitor tests	102
Viewing test results	103
Disabling and re-enabling monitor tests	104
Stopping, restarting, and deleting monitor tests	105
Deleting monitor services	106
Logging	107
Server logs	107
Request logs	108
Audit logs	108
Configuring execution services	109
Managing failover	111
Identifying failover	111
Effects of failover	112
Backing up and restoring Avaki servers	113
Managing the persistent state of GDCs and grid servers	113
Backing up the persistent state of a grid server or a GDC	113
Restoring the persistent state of a grid server or a GDC	114
Managing the persistent state of share servers	115
Backing up a share server	115
Restoring a share server	116
Managing a DGAS cache	117
Clearing cached credentials	117
Deleting files and directories from the cache	119
Saving a copy of the cache	120
Syncing the cache	121
Viewing and modifying cache statistics	123
DGAS properties and cache size	124
Creating and managing CIFS shares	125
Prerequisites	125
Creating a CIFS share	126
Setting system properties	128
System properties on servers and clients	128
Setting system properties for Avaki servers and Avaki Studio	129
Setting system properties for command clients	129
System property reference	131
Setting a server's host name or IP address	131
Configuring read threads for cache services	131
Setting message timeout properties	133

Setting the location of the temp directory	135
java.io.tmpdir	135
Setting cache service properties	135
com.avaki.cache.cacheDir	135
com.avaki.cache.writeInvalidationQueueSize	136
com.avaki.maxActiveCachables	136
com.avaki.result.gcInterval	136
com.avaki.vaultStateCacheSize	137
Setting cache sizes for data service plug-ins	137
com.avaki.dataservice.pluginCacheSize	137
com.avaki.dataservice.styleSheetCacheSize	137
com.avaki.dataservice.styleSheetCachePoolSize	137
Setting share server properties	138
com.avaki.shareServerCircularLinkChecking	138
com.avaki.shareServerThreadPoolMaxSize	138
com.avaki.shareServerThreadPoolMinSize	138
com.avaki.shareReadBufferSize	138
com.avaki.shareReadBufPoolSize	138
Setting the chunk size for query engine sort operations	139
com.avaki.queryEngine.sortChunkSize	139
Setting encryption for grid objects	139
com.avaki.content.encryptionLevel	139
Setting HTTP ports	140
com.avaki.HttpPort	140
com.avaki.HttpsPort	140
Setting HTTP keepalive parameters	140
com.avaki.proxy.httpKeepAliveParams	140
Setting client properties	141
java.protocol.handler.pkgs	141
com.avaki.badPortCacheSize	141
com.avaki.badPortExpiration	141
Setting the remote coherence window for configurations	141
com.avaki.remoteconfig.coherenceWindow	141
Setting the XML indent size	142
com.avaki.generatedXMLIndentSize	142
Setting multiplexing socket properties	142
com.avaki.mux.maxParallelChannels	142
com.avaki.mux.maxWriteChunk	142
com.avaki.mux.sendBufferSize	143
Setting Kerberos properties	143
java.security.krb5.conf	143

java.security.krb5.kdc	143
java.security.krb5.realm	143
Setting the cache size for the virtual database	144
com.avaki.VirtualDbTableCacheSize	144
Setting properties for the remote object stub cache	144
com.avaki.lasInvoker.cacheSize	144
com.avaki.lasInvoker.poolSize	144
Setting properties for the schedule exclusion cache	144
com.avaki.scheduleExclusionCacheSize	145
com.avaki.scheduleExclusionCacheExpiration	145
Setting the operations status expiration interval	145
com.avaki.jobStatusExpiration	145
Setting the page size for LDAP results	145
com.avaki.ldap.resultPageSize	145
Setting the TDS port	146
com.sybase.avaki.tdsPort	146

Chapter 5

Managing authentication services, users, and groups 147

Managing authentication services	148
Integrating LDAP authentication services	148
Configuring Kerberos access	152
Providing Kerberos information	152
Mapping the LDAP DN to a Kerberos user ID	154
Importing user accounts from an LDAP authentication service	157
Users and groups	157
Disabling import on login	157
Importing the user accounts	158
Importing groups from an LDAP authentication service	159
Integrating NIS authentication services	162
Importing user accounts from an NIS authentication service	163
Users and groups	163
Disabling import on login	163
Importing the user accounts	164
Importing groups from an NIS authentication service	165
Viewing existing authentication services	166
Deleting authentication services	166
Managing users	167
Creating grid user accounts	168
Setting up home directories	169
Creating the home directory	169

Setting ACLs on a grid object	171
Changing a grid user's password	175
Mapping Avaki users to database users	176
Adding database identity mappings	177
Viewing database identity mappings	180
Modifying database identity mappings	182
Deleting database identity mappings	183
Keeping authentication services up to date	185
Configuring LDAP authentication services to refresh on a schedule	186
Viewing and modifying user account information	188
Deleting user accounts	190
Managing groups	191
Creating grid groups	191
Adding users to groups	192
Removing users from groups	194
Refreshing imported groups	195
Viewing and modifying group information	195
Deleting groups	198

Chapter 6

Basic tasks 199

Logging in	199
Signing in to a local domain	200
Signing in to a remote domain	202
Accessing CIFS shares	204
Prerequisites	204
Mapping a CIFS share	205
Naming Avaki objects	207
Case sensitivity	207
Character restrictions	207
Managing files and directories	208
Creating grid directories	208
Moving files and directories	210
Renaming files and directories	212
Copying files and directories	214
Making a local copy of a grid file	216
Linking files and directories	217
Deleting files and directories	220
Using and managing categories	221
About categories	221
Browsing categories	222

Creating categories	224
Adding grid objects to a category	226
Removing grid objects from a category	228
Deleting categories	230
Searching the data catalog	231
Creating a search service	231
Performing a search	233
Deleting a search service	236

Chapter 7

Using ACLs and attributes 237

Using access control lists	237
Viewing ACLs	238
Modifying permissions in ACLs	240
Adding users and groups to ACLs	243
Using attributes	246
Viewing attributes	247
Creating new attributes	249
Setting attribute values	253
Deleting attributes	255

Chapter 8

Managing Avaki shares 257

Overview of Avaki shares	258
Creating Avaki shares	258
Refreshing Avaki shares	262
Adding share servers to Avaki shares	263
Removing share servers from Avaki shares	265
Changing the configuration of Avaki shares	266
Changing the refresh schedule	266
Adding entries to the refresh schedule	267
Configuring schedule exclusions	274
Removing entries from the refresh schedule	278
Changing the encryption level	279
Changing the load balancing factor	280
Uploading files to Avaki shares	282
Moving shared data	283
Taking Avaki shares on- and off-line	285
Taking Avaki shares off-line	285

Bringing Avaki shares on-line	286
Disconnecting Avaki shares	288

Chapter 9

Managing interconnections 289

About interconnecting Avaki domains	289
Proxy servers and proxy routing tables	290
The provider domain	290
Is a proxy server needed?	290
Provider tasks	290
The consumer domain	291
Other tasks	291
Connecting to a provider domain	292
Configuring proxy routing tables	292
Setting up the interconnection	293
User access methods	295
Submitting user or group interconnection IDs	295
Sending a user interconnection ID	296
Sending a group interconnection ID	298
Enabling interconnection access	299
Configuring a proxy server	300
Setting the host name	300
Using nondefault ports	300
Proxy server configuration	300
Enabling user or group access	305
Exposing users in a two-way interconnection	309
Viewing interconnected domains	311
Enabling cross-domain messaging	312
Disabling cross-domain messaging	313
Disconnecting interconnected domains	315

Appendix A

Setting up log properties files 317

Server log properties files	317
Avaki logging categories	318
Configuring audit logging	320
Events captured by audit logging	322
Sample log properties file	323
Request log properties files	328

Appendix B*DGAS access effects* **331**Host admission policy **332**Mapping overrides in a DGAS **333**Default user, group, UID and GID **333**Changing permissions and ownership **334**DGAS mapping cache **336**Client attribute caching **336**Access using older NFS clients **337**CIFS-based access **338****Appendix C***Upgrading to Avaki 7.0* **339**Interoperability **339**Preparing to upgrade **340** Upgrade planning **340** Preparing your domain for the upgrade **341**Upgrade procedures **343** Copying internal data **343** Starting the upgraded servers **346**Post-upgrade tasks **346**Testing **347***Glossary* **349**

Preface

This *Sybase Avaki EII Administration Guide* explains how to install and deploy an Avaki domain on computers in an existing communication network.

Note This book and the product's user interfaces refer to Sybase Avaki EII software as *Avaki* or *Avaki Data Grid*.

Audience

This guide is intended for anyone who plans Avaki deployments, installs Avaki software, or administers an Avaki domain. Readers are expected to have:

- Knowledge of and administrative access to the hardware and operating systems on which they are installing Avaki software
- Knowledge of their network's topology, including how and where firewalls are configured

Organization

This book is organized as follows:

Chapter 1 Planning your Avaki deployment	Lists requirements and recommendations to help you prepare to deploy an Avaki domain.
Chapter 2 Installing Avaki software	Describes how to load Avaki software (for all Avaki server types) onto a machine and how to initialize a grid domain controller.
Chapter 3 Setting up Avaki servers and clients	Explains how to add servers and clients to a data grid and how to configure them.
Chapter 4 Managing Avaki servers	Explains how to set up server monitoring, how to set up a CIFS share, how to manage DGAS internal caches, and other administrative tasks.
Chapter 5 Managing authentication services, users, and groups	Shows how to integrate external authentication services with an Avaki domain, how to import, create, and delete user accounts and groups, and how to change passwords.
Chapter 6 Basic tasks	Explains how to sign in to an Avaki domain, manipulate Avaki directories, categories, and files (including creating links), and set up and perform searches.
Chapter 7 Using ACLs and attributes	Explains how to configure permissions and attributes for files, directories, services, and other Avaki objects.
Chapter 8 Managing Avaki shares	Shows how to share your local files and directories into an Avaki domain, schedule rehashes, upload files to shares, and disconnect and disable shares.
Chapter 9 Managing interconnections	Tells you how to interconnect two Avaki domains.
Appendix A Setting up log properties files	Describes the format and content of properties files that control logging for Avaki servers.
Appendix B DGAS access effects	Explains how a DGAS accesses data when it receives requests from NFS or CIFS clients.
Appendix C Upgrading to Avaki 7.0	Describes the steps for upgrading an Avaki domain to the current software release.
Glossary	Defines terms used in this guide.

Related documentation and online help

Manuals

These manuals make up the Avaki documentation set:

- *Sybase Avaki EII Overture*
- *Sybase Avaki EII Administration Guide* (includes installation instructions)
- *Data Integration with Sybase Avaki Studio*
- *Sybase Avaki EII Provisioning and Advanced Data Integration Guide*
- *Sybase Avaki EII API Guide*
- *Sybase Avaki EII Command Reference*

The manuals are included, in PDF format, on the CD with the Avaki software. They are stored in the docs subdirectory of the Avaki installation directory.

To access the manuals via Avaki's web user interface, log in to your Avaki domain and click the **Help** link at the top right corner of any page of the web UI.

Online help

In addition to the manuals, Avaki provides online help for commands. To display a list of Avaki commands with brief descriptions, log in to Avaki Data Grid and enter **avaki help**:

```
% avaki help
Avaki Command Line Version <Avaki-Build-20921>
List of domain commands:
  attribute
  backup
  cache
  cat
  categories
  cd
  chmod
  chown
  client
  cp
  dataservice
  dbconn
  dbop
```

dgas
directory
domain
executionservice
file
group
help
id
ldap
ln
locks
login
logout
ls
mkdir
monitor
mv
nis
passwd
patch
permissions
plugin
proxy
pwd
replica
rm
scheduleexclusion
search
security
server
share
shell
sqlview
status
systemproperty
upgrade
user
view
virtualdatabase
virtualschema
whoami

To display a description and syntax for a particular command, enter a command of the form **avaki help** *<command>*. For example:

```
% avaki help mv
usage: avaki mv [--force | -f] <source-grid-path>
<target-grid-path>
```

Description: Move or rename a grid directory or a file in a grid directory. Similar to the Unix mv command.

Conventions

This section describes text conventions used in this guide to represent elements of commands and screen displays.

Command syntax conventions

This table describes conventions that this book uses in command syntax statements. The “Enter this” column tells you whether you need to enter the characters when you type a command. The examples in the “Examples” column are not necessarily complete commands.

Convention	Description	Enter this?	Examples
[]	Square brackets surround optional arguments.	no	avaki login [<user-id>]
{ }	Curly brackets surround groups of required arguments.	no	avaki chmod {--allow --deny --unset}
{ }	Vertical bars separate alternative options within square or curly brackets. If the brackets are square, you need not enter any of the options; if the brackets are curly, you must choose one of the options.	no	avaki backup {--snapshot --recover} avaki share --create [--background --bg]

Convention	Description	Enter this?	Examples
< >	Angle brackets surround placeholder arguments that you must replace with a value such as a path or file name. Square brackets outside the angle brackets indicate that the placeholder is optional.	no	avaki help [<command-name>] avaki help share
*	An asterisk follows an argument that can be entered zero or more times.	no	avaki plugin --generate [--input=<stream-spec>*] avaki plugin --generate --input="name=input1;type=XML"
+	A plus sign follows an argument that can be entered one or more times. Use spaces to separate the values.	no	avaki cat <grid-path>+ avaki cat /home/fred/file1 /home/fred/file2
-	Enter a hyphen or minus sign before a single-letter command option.	yes	avaki mkdir -p
=	Enter an equal sign before the value of an option.	yes	avaki login --auth-service= <auth-service-name>
(space)	A space separates multiple arguments.	yes	avaki cat file1 file2 file3

Conventions for screen examples

This table describes conventions used in examples of user input and system output.

Convention	Description	Example
\$ or C:>	The command prompt	\$ or C:>
< >	A placeholder; replace the text inside the brackets with an option or value	\$ avaki ls <grid-path>
screen font	Text that appears on the screen	sample text
bold screen font	User input—commands that you enter	\$ avaki ls

How to contact Avaki support at Sybase, Inc.

For general information about Sybase technical support, see the *Customer Service Reference Guide* at

<http://www.sybase.com/support/aboutsupport/guide/csrg>

Please contact us with any questions or difficulties you encounter.

By telephone

In North America, call toll free: 1-800-8SYBASE

Outside North America, follow the link below to see a list of Sybase offices and phone numbers around the world.

<http://www.sybase.com/contactus/support>

On the web

If you are a designated contact for a technical support plan, you can log and track cases on the web using the Case Express application. At www.sybase.com, mouse over the **Support and Services** tab and select **Case Management** from the dropdown list. Use the email address and password for your mysybase account to log in.

Planning your Avaki deployment

This chapter lists requirements and recommendations that you should review well in advance of installing Avaki software. These topics are covered:

- [“Planning steps,”](#) below
- [“System requirements”](#) on page 3
- [“Planning use of network ports”](#) on page 8
- [“Planning for SSL certificates”](#) on page 10
- [“Configuring user accounts”](#) on page 11

For introductory information about Avaki, see the *Sybase Avaki EII Overture* manual.

Planning steps

Complete these tasks as you plan an Avaki deployment.

1. With assistance from a Sybase professional services consultant, determine the following:
 - How many of each type of Avaki server do you need in your Avaki domain?
 - Which machines will the Avaki servers run on?

On both questions, be sure to consult people in your organization who are knowledgeable about the requirements of the users and the applications that will use the Avaki grid.

2. Verify that the candidate machines for the Avaki domain meet the hardware and software requirements listed in the section [“System requirements” on page 3](#).
3. Check [“Planning use of network ports” on page 8](#) to find out whether ports used by Avaki conflict with other port usage on your machines. (Most such conflicts can be resolved by directing Avaki servers to use nondefault ports; instructions for doing so appear in the set-up instructions for Avaki servers in [Chapter 3, “Setting up Avaki servers and clients”](#).)
4. Buy SSL certificates if necessary—see [“Planning for SSL certificates” on page 10](#).
5. Set up a DNS alias for your Avaki grid domain controller (GDC)—see [“Creating a DNS alias for the GDC machine” on page 11](#).
6. Set up user accounts for Avaki on local machines or in your directory service—see [“Configuring user accounts” on page 11](#).

Note If you’re upgrading from an earlier version of Avaki software, see the upgrade procedure in [Appendix C, “Upgrading to Avaki 7.0”](#).

System requirements

Before installing Avaki software, make sure that the machines that will host your Avaki domain meet the following hardware and software requirements. Specifications are included for grid domain controllers (GDCs), grid servers, data grid access servers, proxy servers, share servers, command-line clients, NFS clients, CIFS clients, and Avaki Studio.

Category	Requirement	Notes
Platform/Operating system Note: Only US English versions (en_US) are supported.	<p>For servers (all types) and command clients:</p> <ul style="list-style-type: none"> • Intel/Windows 2003 Server, Windows XP Professional • Intel/Red Hat Enterprise Linux ES 3.0 or 4.0 • Intel/SuSE Linux 9.1 or 9.2 • IBM AIX 5.2 maintenance level 01 or 5.3 • SPARC/Solaris 9 or 10 <p>For Avaki Studio: Windows 2003 Server, Windows XP Professional</p> <p>For NFS clients: NFS version 2 or version 3 client software</p> <p>For CIFS clients: Windows 2003 Server, Windows XP Professional</p>	<p>In this documentation set, we refer collectively to all the Unix-like operating systems we support (including Linux, AIX, and Solaris) as “Unix.”</p> <p>Contact Sybase for information about using other operating systems with Avaki software.</p>
Minimum memory (RAM) Note: These specifications are for memory in addition to that used by the OS and any other processes running on the machine.	<p>For all platforms:</p> <ul style="list-style-type: none"> • Grid domain controllers: 512MB • Grid servers: 512MB • Share servers: 256MB • Data grid access servers: 1GB • Proxy servers: 256MB • Command clients: 256MB • Avaki Studio: 256MB 	<p>For descriptions of Avaki servers, see the Glossary (page 349).</p> <p>For grid servers or GDCs with large data processing loads (heavy database activity, many web services, etc.), consider increasing memory.</p>

Category	Requirement	Notes
Disk drives	<p>We recommend two SCSI or IDE/SATA disk drives for each Avaki server: one for the operating system and Avaki log files, the other for Avaki software and data.</p> <p>For grid domain controllers, grid servers, data grid access servers, and share servers, disks should be 10K RPM minimum. For heavy loads: 15K RPM disks or 15K RPM disk arrays with RAID 5.</p> <p>For Avaki Studio, proxy servers, and command clients, slower disks may be adequate.</p>	
Minimum free disk space	<p>For servers on all platforms:</p> <ul style="list-style-type: none"> • Grid domain controllers: 512MB • Grid servers: 512MB, or 120MB plus about 1K per shared file • Share servers: 200MB • Data grid access servers: 120MB • Proxy servers: 120MB • Command clients: 120MB <p>For Avaki Studio: 200MB</p>	<p>Software installation requires free disk space equal to about twice the size of the installer image, or 600 to 700MB.</p> <p>All Avaki servers and clients require additional disk space to store log files and state databases. Grid servers need further additional disk space; the amount varies based on how heavy a data processing load the grid server carries and how many shared files it supports.</p> <p>Grid servers and data grid access servers also need disk space for caches. The space needed depends on the amount of data to be cached.</p>
Recommended free disk space	<p>For each grid domain controller, grid server, and data grid access server, we suggest at least 20GB of free disk space.</p> <p>For Avaki Studio: 300MB</p>	

Category	Requirement	Notes
Web browsers	<p>To run the Avaki web user interface, one of the browsers listed here must be able to access a machine in the Avaki domain.</p> <p>For all supported versions of Windows:</p> <ul style="list-style-type: none"> • Firefox 1.5.0.1 or later • Microsoft Internet Explorer 6.x (not IE 7) • Netscape 7.0 <p>For Red Hat ES 3: Mozilla 1.7.8 For Red Hat ES 4: Firefox default</p> <p>For SuSE Linux 9.1, 9.2 or 9.3: Konqueror 3.2.1 or later</p> <p>For Solaris 9: Netscape default For Solaris 10: Mozilla default</p> <p>The Avaki web UI is based on web standards (HTML, CSS, JavaScript) so it may be usable through other OS/browser combinations. However, only those listed above are supported.</p>	<p>For OS platforms that are distributed with browsers, Avaki supports only the included browser.</p> <p>If you have installed the Windows 832894 security update (MS04-004) or the 821814 hotfix, you may find that your web browser sometimes drops POST parameters. (The login page in the Avaki UI loops back to itself, and submitting various forms gives a “Page not found: null” error.) The following link provides an update that corrects the problem:</p> <p>http://support.microsoft.com/default.aspx?kbid=831167</p>
Databases	<p>You can use Avaki database tools with several relational database management systems, including IBM DB2, MySQL, Microsoft SQL Server, Oracle, Sybase ASA, Sybase ASE, and Sybase IQ.</p>	<p>See the table that begins on page 6 for information on tested DBMS versions and drivers.</p>
Application development	<p>To write programs that access Avaki using JDBC, you need:</p> <ul style="list-style-type: none"> • For IBM AIX: JRE 1.4.2 or later • For all other platforms: JRE 1.4.2_07 or later <p>To write data service plug-ins in Java, you need:</p> <ul style="list-style-type: none"> • A Java compiler such as Sun JDK 1.4 • Apache Ant 1.5.3.1 or later 	<p>In an Avaki domain with primary and secondary GDCs, install a UPS on each GDC machine.</p>
Other equipment	<p>Recommended: Uninterruptable power supply (UPS) on the grid domain controller machine(s)</p>	<p>In an Avaki domain with primary and secondary GDCs, install a UPS on each GDC machine.</p>

Databases and JDBC drivers

The following table provides information on DBMS versions and JDBC drivers that have been tested for connecting Avaki to back-end databases.

Note With any DBMS, your JDBC driver must support JDBC 2.0 or later.

IBM DB2

DB2 v8.1.0.36 with driver IBM DB2 JDBC Universal Driver 1.0

Database driver class	COM.ibm.db2.jdbc.net.DB2Driver
------------------------------	--------------------------------

Connection string	jdbc:db2://<host-name>:<port>/<db-name>*
--------------------------	--

Microsoft SQL Server

SQL Server 2000 with SQL Server 2000 Driver for JDBC Service Pack 1

Database driver class	com.microsoft.jdbc.sqlserver.SQLServerDriver
------------------------------	--

Connection string	jdbc:microsoft:sqlserver://<host-name>:<port>;DatabaseName=<db-name>*
--------------------------	---

MySQL

MySQL 5.0 with MySQL Connector/J driver 5.0

Database driver class	com.mysql.jdbc.Driver
------------------------------	-----------------------

Connection string	jdbc:mysql://<host-name>:<port>/<db-name>*
--------------------------	--

Oracle 10g

Oracle 10g with Thin JDBC driver

Database driver class	oracle.jdbc.driver.OracleDriver
------------------------------	---------------------------------

Connection string	jdbc:oracle:thin:@<host-name>:<port>:<db-name>*
--------------------------	---

Sybase ASA

Sybase ASA 9 with driver jConnect 5.5 or higher. JConnect 6.x is recommended.

Database driver class	For jConnect 5.x: com.sybase.jdbc2.jdbc.SybDriver For jConnect 6.x and above: com.sybase.jdbc3.jdbc.SybDriver
------------------------------	--

Connection string	jdbc:sybase:Tds:<host-name>:<port>/[<db-name>]*
--------------------------	---

* Replace <host-name>, <port>, and <db-name> with the name of the database host, the port number, and the database name. The default port for jConnect (Sybase databases) is 15000.

Sybase ASE

Sybase ASE 12.5.x and 15 with driver jConnect 5.5 or higher. JConnect 6.x is recommended.

Database driver class	For jConnect 5.x: com.sybase.jdbc2.jdbc.SybDriver For jConnect 6.x and above: com.sybase.jdbc3.jdbc.SybDriver
Connection string	jdbc:sybase:Tds:<host-name>:<port>/[<db-name>] ^{*†‡}

Sybase IQ

Sybase IQ 12.6 with driver jConnect 6 or higher. JConnect 6.x is recommended.

Database driver class	com.sybase.jdbc3.jdbc.SybDriver
Connection string	jdbc:sybase:Tds:<host-name>:<port>/[<db-name>] ^{**‡}

- * Replace <host-name>, <port>, and <db-name> with the name of the database host, the port number, and the database name. The default port for jConnect (Sybase databases) is 15000.
- † The CHARSET JDBC property may be required if you're connecting across architectures (e.g. connecting a Windows JDBC client to a Linux ASE instance). For example, jdbc:sybase:Tds:<host-name>:<port>/<db-name>?CHARSET=iso_1.
- ‡ If you use jConnect 5.x with IQ or ASE, set the FAKE_METADATA property to true. For example, jdbc:sybase:Tds:<host-name>:<port>/<db-name>?FAKE_METADATA=true.

Planning use of network ports

To identify conflicts with your current port usage, review this list of the default TCP ports used by Avaki servers. If you find any conflicts, identify alternate port numbers for your Avaki servers to use. Instructions on configuring servers to use nondefault ports appear in [Chapter 3, “Setting up Avaki servers and clients”](#). The ports listed here are all for incoming traffic.

Server type	Port no.	Description
Grid server/GDC	1098	RMI port
	3099	Connect port (JNDI port)
	3100*	RMI registry port. This port number is always one higher than the connect port number.
	3101*	RMI server port. This port number is always two higher than the connect port number.
	3095	Apache Axis port for JBoss/SOAP monitoring.
	4444	Internal RMI/SSL port
	7080*	HTTP listening port
	7083	HTTP port for RMI class loading
	8089	RMI port for JMS
	8090	OIL port for JMS
	8092	OIL2 port for JMS
	8093	OIL2 port for JMS
	8443*	HTTPS listening port (SSL port)
	8511*	SSL port for the share server protocol used between Avaki servers
	8512*	Clear socket port for the share server protocol used between Avaki servers
15000	TDS port used by Java applications to communicate with Avaki via Sybase JDBC or ODBC drivers	

* Ports used for remote communication: if you're setting up a grid domain that crosses a firewall, open these ports.

Server type	Port no.	Description
Share server	2098	RMI port
	2099	Connect port (JNDI port)
	2100 [*]	RMI registry port. This port number is always one higher than the connect port number.
	2101 [*]	RMI server port. This port number is always two higher than the connect port number.
	6444	Internal RMI/SSL port
	9080	HTTP listening port
	9083	HTTP port for RMI class loading
	9443	HTTPS listening port (SSL port)
	9510 [*]	Clear socket port for the share server protocol used between Avaki servers
	9511 [*]	SSL port for the share server protocol used between Avaki servers
Data grid access server (DGAS)	<i>varies</i> [*]	Mount protocol port, a free port chosen by the OS. (When the DGAS restarts, it continues to use the same port.)
	<i>varies</i> [*]	NLM protocol port, a free port chosen by the OS. (When the DGAS restarts, it continues to use the same port.)
	137 – 139 [*]	Ports for SMB over NetBIOS
	445 [*]	Port for SMB over TCP/IP
	1399 [*]	Connect port (RMI registry port)
	1599 [*]	RMI server port
	2049	NFS protocol port

^{*} Ports used for remote communication: if you're setting up a grid domain that crosses a firewall, open these ports. NFS, SMB, NLM and mount ports may not need to be exposed through a firewall—in a typical deployment, the DGAS and its clients are on the same side of the firewall.

Server type	Port no.	Description
Proxy server	1199*	Connect port (RMI registry port)
	18080*	HTTP listening port
	18443*	HTTPS (SSL) listening port
	38080*	RMI server port

* Ports used for remote communication: if you're setting up a grid domain that crosses a firewall, open these ports.

Planning for SSL certificates

Avaki servers and GDCs use SSL certificates to ensure secure data transmission; generally one certificate is required for each IP address at which an Avaki server is running. (The exception is data grid access servers—DGASes do not require SSL certificates.) You must choose one of the following options for the Avaki servers in your grid domain:

- Trust Avaki's self-signed SSL certificates, which are included with each installation. This is a reasonable choice if your data grid will be operated and accessed by a single organization. If you choose this option, however, each grid user will have to accept the SSL certificate at each login.
- Use the Java **keytool** utility to generate your own self-signed SSL certificates. This is also a good choice if your data grid will be operated and accessed by a single organization. If you choose this option, you must follow the steps in [“Generating self-signed SSL certificates” on page 34](#) for each Avaki server.
- Obtain and install SSL certificates from a trusted certificate authority. Buying certificates is a good choice if your data grid will be accessed by more than one organization. If you choose this option, you must purchase certificates from one of the root certification authorities that are recognized by your web browser—for example, VeriSign, Inc. (verisign.com) or Thawte, Inc. (www.thawte.com). We recommend that you contact the certification authority four to eight weeks before you plan to install Avaki software; it can take some time to complete the paperwork. Be sure to complete the purchase *before* you install Avaki software.

Creating a DNS alias for the GDC machine

Changing the host name of the grid domain controller (GDC) can prevent other Avaki servers in the domain from communicating with the GDC. To avoid this problem, we recommend that you set up a DNS alias for the machine on which your GDC will run—“AvakiGDC,” for example. The alias lets you change the name of the GDC machine or move the GDC to another machine without disrupting your Avaki domain.

Configuring user accounts

To set up an Avaki domain, you need access to at least two native (nonAvaki) user accounts:

- One account with administrative privileges
- One account without administrative privileges. You might find it convenient to establish a dedicated nonadministrative account for use by data grid administrators. We refer to this account as local user avaki.

Make sure that both accounts are accessible from each of the machines that will host Avaki grid domain controllers, servers, or command clients. The accounts can be configured on the machines or in a directory service such as Active Directory or NIS.

In Unix, a user who has administrative privileges typically logs in as root; in Windows, a user who has administrative privileges is listed in the Administrators group. A user account with administrative privileges is required in order to run certain Avaki commands, but for security reasons, an account with administrative privileges should not be used to run most servers in a grid domain. (Exceptions are data grid access servers, and in some circumstances, grid servers and share servers. See [“User accounts for servers supporting Avaki shares,”](#) below, for more information.) In Windows, configure the account that does not have administrative privileges (local user avaki) so that the user has permission to log on as a service.

Note For information on Avaki user accounts, see [“Setting up administrative accounts in Avaki”](#) on page 45 and [“Managing users”](#) on page 167.

User accounts for enabling auto-restart

Avaki provides an automatic restart feature for Avaki servers and grid domain controllers. When auto-restart is enabled, the Avaki server or GDC restarts automatically when the computer restarts. If auto-restart is not enabled, you must bring the GDC or server back up manually each time its machine reboots. If you plan to use Avaki's auto-restart feature, make sure user accounts are available as follows:

- On a Unix machine, to set up an Avaki server or GDC with auto-restart enabled, you must be logged in to the machine as a user who has administrative privileges.
- On a Windows machine, to set up an Avaki server or GDC with auto-restart enabled, you must be logged in to the machine as a user who has permission to log on as a service.

User accounts for servers supporting Avaki shares

If you want an Avaki share server or grid server to provide write access to its shared directories, the share server or grid server must run under an operating system account that has write access to the underlying file system. (In the rest of this section, the term *share server* refers to either case—a dedicated share server or the local share server function of a grid server.)

On Unix systems (including Linux), a share server that provides write access typically runs under the root account. Note that on most Unix systems the root account has full access to local file systems, but not to NFS-mounted file systems. In order for root to have full access to NFS-mounted file systems, the NFS server for each of these file systems may have to be configured to allow remote NFS root access from the share server—see your operating system documentation for details.

On Unix systems, Avaki share servers try to avoid creating new files under the root user account on the remote file system by automatically making the owner of each file created via Avaki the same as the owner of the OS directory in which it is created. For example, suppose `/home/joeuser` is shared into the data catalog via a share server running as the OS user `root` and is owned by the OS user `joeuser`. If a file called `/home/joeuser/myfile` is created via the data grid, it will be owned by the user `joeuser`, not by `root`. There is no comparable functionality for ownership adjustment when Avaki is sharing data out of a Windows file system.

Note Be sure to review the readme file for this release. You can find it in `<Avaki-install-dir/docs`.

Installing Avaki software

Read this chapter to learn how to:

- Install Avaki software for any Avaki server.
- Start the first Avaki server in your domain, the grid domain controller (GDC).

Before you install Avaki software, see [Chapter 1, “Planning your Avaki deployment”](#).

For introductory information about Avaki, data grids, and enterprise information integration (EII), see the *Sybase Avaki EII Overture*.

In this chapter:

- [“Overview of the installation process,”](#) below
- [“Loading and starting Avaki Data Grid software”](#) on page 15
- [“Setting up administrative accounts in Avaki”](#) on page 45

Overview of the installation process

The procedure for deploying an Avaki grid domain is as follows:

1. Perform planning steps as described in [Chapter 1, “Planning your Avaki deployment”](#).
2. Copy the Avaki installer from the Web or CD onto the machine that will be your primary grid domain controller (GDC). See [“Loading and starting Avaki Data Grid software” on page 15](#).
3. Install Avaki software on the primary GDC machine. If you plan to have a secondary GDC, install on that machine as well.
4. (Optional) Specify a host name or IP address for your GDC to advertise if the default (the server machine’s name) is unsuitable. See [“Setting a server’s host name or IP address” on page 32](#).
5. (Optional) Generate your own SSL certificates if you’ve chosen this option. See [“Generating self-signed SSL certificates” on page 34](#).
6. Start the primary GDC. See [“Starting a primary GDC” on page 37](#).
7. Install an SSL certificate on the GDC machine. See [“Installing an SSL certificate” on page 39](#).
8. Configure the primary GDC. See [“Configuring a primary GDC” on page 41](#).
9. If you have a secondary GDC, start it next. See [“Starting a secondary GDC” on page 43](#).
10. Copy and install Avaki software on the machines that will host grid servers, data grid access servers, share servers, proxy servers, command clients, and Avaki Studio.
11. Configure and start the servers and clients. See [Chapter 3, “Setting up Avaki servers and clients” on page 47](#).
12. Perform additional administrative tasks, such as adding users and setting permissions. See [Chapter 5, “Managing authentication services, users, and groups” on page 147](#).
13. (Optional) If you installed Avaki Studio, start it and set it up. See *Data Integration with Sybase Avaki Studio*.

Note If you’re upgrading from an earlier version of Avaki software, do not use the installation procedures. See the upgrade procedure in [Appendix C, “Upgrading to Avaki 7.0”](#).

Loading and starting Avaki Data Grid software

This section describes how to

- Run the installer that loads Avaki software for all installation types (grid domain controllers, servers, clients, and Avaki Studio)
- Start the base Avaki grid on one or two machines. The first server is called the grid domain controller (GDC). If you wish, you can also start a secondary GDC to provide failover support.

Choose machines for the GDCs that meet the hardware and software requirements described in this section and in [“System requirements” on page 3](#). You must have an account on each machine, and you must install the Avaki binaries.

Each GDC should be installed on a dedicated, fail-safe machine with a fixed IP address. The machines must be dedicated because the performance of the Avaki grid depends on the machine’s available resources. If other CPU- or memory-intensive applications are running on the GDC, the grid’s responsiveness might slow considerably. The machine should not be available to casual users.

Note Be sure to review the readme file for this release. You can find it in <Avaki-install-dir/docs.

These sections describe how to load and start GDCs:

- [“Installing in Unix,”](#) below
- [“Installing in Windows” on page 22](#)
- [“Generating self-signed SSL certificates” on page 34](#)
- [“Starting a primary GDC” on page 37](#)
- [“Installing an SSL certificate” on page 39](#)
- [“Configuring a primary GDC” on page 41](#)
- [“Starting a secondary GDC” on page 43](#)

Installing in Unix

For machines running any supported variety of Unix (including Linux), the following sections describe installing Avaki software, starting a primary grid domain controller, and logging in as an Avaki administrator:

- [“Before installing on a system running any version of Unix,”](#) below
- [“Installing Avaki software”](#) on page 16
- [“Starting a primary GDC”](#) on page 37

Before installing on a system running any version of Unix

The Avaki installer requires certain X Window System libraries. Generally, these libraries are packaged with the operating system, but we recommend that you check for them before you install Avaki on any system running Red Hat or SuSE Linux, IBM AIX, or Sun Solaris. The required X11R6 libraries are:

- libXp.so.6
- libXt.so.6
- libXext.so.6
- libXtst.so.6
- libX11.so.6
- libSM.so.6
- libICE.so.6

Installing Avaki software

On a machine running Unix, follow these steps to perform any type of installation, including a grid domain controller, grid server, data grid access server, share server, firewall proxy server, or command client.

- Step 1** Check the space available in the /tmp directory on the target machine. The Avaki installer requires free disk space equal to about twice the size of the installer image. If the space is not available in the /tmp directory, set the IATEMPDIR environment vari-

able to point to a location where sufficient space is available. However, note the following:

- If the IATEMPDIR location is on the same disk where you are installing Avaki software, the installer needs space equal to at least *three* times the size of the installer image.
- The installer uses space in /tmp even when IATEMPDIR points elsewhere.
- If /tmp becomes full during installation, whether IATEMPDIR is set or not, the result might be an incomplete installation.

Step 2 Run `install.bin` in the Avaki installation directory to start the Avaki installer for your operating system. You'll see this screen:

```

Preparing to install...
Extracting the JRE from the installer archive...
Unpacking the JRE...
Extracting the installation resources from the installer archive...
Configuring the installer for this system's environment...

Launching installer...

Preparing CONSOLE Mode Installation...

=====
                                (created with InstallAnywhere by Zero G)
-----

=====
Introduction
-----

InstallAnywhere will guide you through the installation of Avaki Data Grid 7.0.

It is strongly recommended that you quit all programs before continuing with
this installation.

Respond to each prompt to proceed to the next step in the installation.  If you
want to change something on a previous step, type 'back'.

You may cancel this installation at any time by typing 'quit'.

PRESS <ENTER> TO CONTINUE:

```

Step 3 Press **Enter** to proceed with the installation. The next screen offers you a choice of product features.

```
=====
Choose Product Features
-----
```

```
ALL PRODUCT FEATURES ARE SELECTED BY DEFAULT.  PRESS <RETURN> TO INSTALL ALL
FEATURES.  TO DESELECT FEATURES,  ENTER A COMMA SEPARATED LIST OF NUMBERS
REPRESENTING THE FEATURES YOU WOULD LIKE TO DESELECT AND THEN PRESS <RETURN>.
```

```
TO VIEW A FEATURE'S DESCRIPTION,  ENTER '?<NUMBER>'.
```

- ```
1- [X] Grid Server
2- [X] Data Grid Access Server
3- [X] Share Server
4- [X] Firewall Proxy Server
5- [X] Command-line client
```

```
Press <RETURN> or deselect features:
```

```
: 3,4,5
```

**Step 4** You must deselect from the list of components—that is, enter the number or numbers of the Avaki components you *do not* want to install, separated by commas. For example, to install only a grid domain controller or a grid server, enter **2,3,4,5**. To install a grid domain controller and a data grid access server, enter **3,4,5**.

The grid components are as follows:

- **Grid Server:** A workhorse server that maintains a portion of the grid domain's namespace, runs grid services such as shares and searches, and allows you to execute Avaki commands. A grid server can also be configured as a GDC.
- **Data Grid Access Server:** Enables access to the grid domain via NFS and CIFS clients
- **Share Server:** Provided dedicated support for linking data into the grid domain
- **Firewall Proxy Server:** Supports multisite communications across firewalls
- **Command-line client:** Command line tools only. These are primarily for administrators and advanced users.

```
=====
License Agreement

```

Installation and use of Avaki Data Grid 7.0 requires acceptance of the following License Agreement:

```
LICENSE AGREEMENT
```

```
United States
```

```
20060501
```

IMPORTANT NOTICE: Read this License Agreement ("Agreement") carefully before using the enclosed Program. You may Use the Program acquired in the United States and Canada only, and only in accordance with the following terms and conditions. IF YOU DO NOT AGREE TO BE BOUND BY THESE TERMS, YOU MAY NOT USE THE PROGRAM. By downloading, installing, or using the Program in any way, You acknowledge that You have read, understand and agree to the terms of this Agreement. If You do not agree with these terms, present your receipt or other proof of purchase, together with the Program media, Documentation and packaging (if any) to the entity from which You obtained this product within 30 days to request a refund. THIS IS A LICENSE AND NOT A SALE.

If You, Your company, or Your public agency have signed a written software license agreement with Sybase, Inc., or Sybase Canada Limited ("Sybase") or a Sybase subsidiary or authorized reseller, covering the Use of the enclosed Program, the terms of the signed license agreement shall take precedence over any conflicting terms of this Agreement.

```
PRESS <ENTER> TO CONTINUE:
```

**Step 5** Press **Enter** to display the next portion of the license agreement. Keep pressing **Enter** until you see this prompt:

```
DO YOU ACCEPT THE TERMS OF THIS LICENSE AGREEMENT? (Y/N): y
```

**Step 6** Enter **y** to accept the license agreement and proceed with the installation.

```
=====
Important: Port Conflicts

```

```
Please read before continuing:
```

```
Check for Port Conflicts:
```

Each Avaki server uses a set of TCP ports to listen for incoming traffic. If another application on your machine uses an Avaki port, the Avaki server will not run properly. After installing, see the Avaki Administration Guide for a list of default ports and instructions on reassigning ports in case of conflicts. (Avaki command clients and Avaki Studio use no listening ports.)

IMPORTANT INFORMATION COMPLETE. PRESS <ENTER> TO CONTINUE:

**Step 7** Press **Enter** to proceed.

=====  
Choose Install Folder  
-----

Where would you like to install?

Default Install Folder: /root/AvakiDataGrid70

ENTER AN ABSOLUTE PATH, OR PRESS <ENTER> TO ACCEPT THE DEFAULT

:

**Step 8** Press **Enter** to accept the default installation directory, or enter the path to another directory.

**Caution** If you are upgrading to a new release of Avaki software, install the new release in a new directory. (Accepting the installer's default directory is a good choice.) If you install the new release in the same directory as the older release, the upgrade will fail. (See [Appendix C, "Upgrading to Avaki 7.0"](#) for instructions on performing an upgrade.)

=====  
Choose Link Location  
-----

Where would you like to create links?

- >1- Default: /root/AvakiDataGrid70
- 2- In your home folder
- 3- Choose another location...
  
- 4- Don't create links

ENTER THE NUMBER OF AN OPTION ABOVE, OR PRESS <ENTER> TO ACCEPT THE DEFAULT

:



**Step 9** Choose a directory for links to the scripts for starting Avaki servers and clients.

```
=====
Pre-Installation Summary

```

Please Review the Following Before Continuing:

Product Name:

Avaki Data Grid 7.0

Install Folder:

/root/AvakiDataGrid70

Link Folder:

/root/AvakiDataGrid70

Product Components:

Grid Server,  
Data Grid Access Server

Disk Space Information (for Installation Target):

Required: 260,575,330 bytes

Available: 39,606,161,408 bytes

PRESS <ENTER> TO CONTINUE:

**Step 10** Check the details in the Pre-Installation Summary. Press **Enter** to install, or type **back** to correct information that you entered previously.

```
=====
Installing...

```

```
[===== | ===== | ===== | =====]
[----- | ----- | ----- | -----]
```

```
=====
Installation Complete

```

Congratulations. Avaki Data Grid 7.0 has been successfully installed to:

/root/AvakiDataGrid70

PRESS <ENTER> TO EXIT THE INSTALLER:

#

When the installation is complete, press **Enter** and the command prompt reappears.

Skip to [“How to proceed” on page 31.](#)

## Installing in Windows

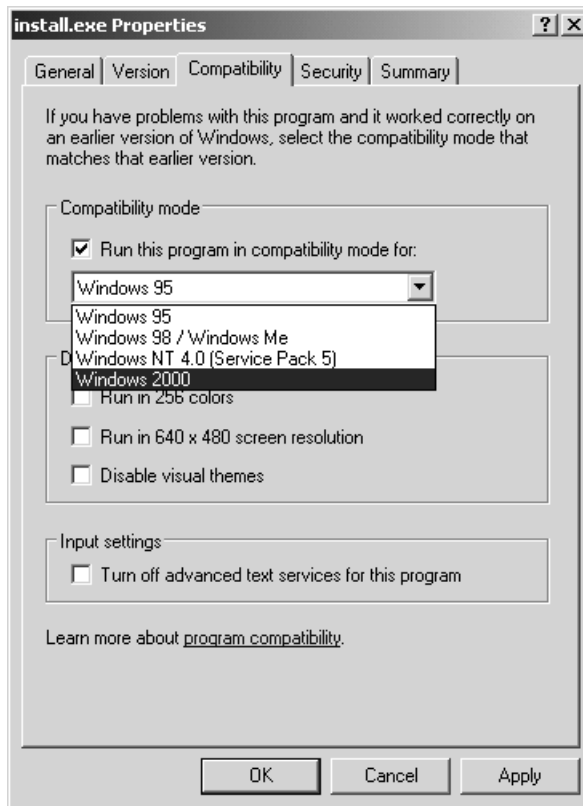
If you’re installing on a machine running Windows XP, skip to the [“Windows installation steps” on page 24.](#)

### Preinstallation steps for Windows 2003

Follow these steps to enable the Avaki installer to run in Windows 2003.

- Step 1** Copy the Avaki installer, `install.exe`, from the CD or the web to a location on the hard disk of the Windows 2003 machine. This should be on a local disk.
- Step 2** In Windows Explorer, navigate to the location to which you copied the `install.exe` file, right-click `install.exe`, and select **Properties** from the menu.

**Step 3** In the resulting dialog box, `install.exe` Properties, click the **Compatibility** tab.



**Step 4** Under Compatibility mode, click the box to check “Run this program in compatibility mode for”.

**Step 5** Select **Windows 2000** from the pull-down menu.

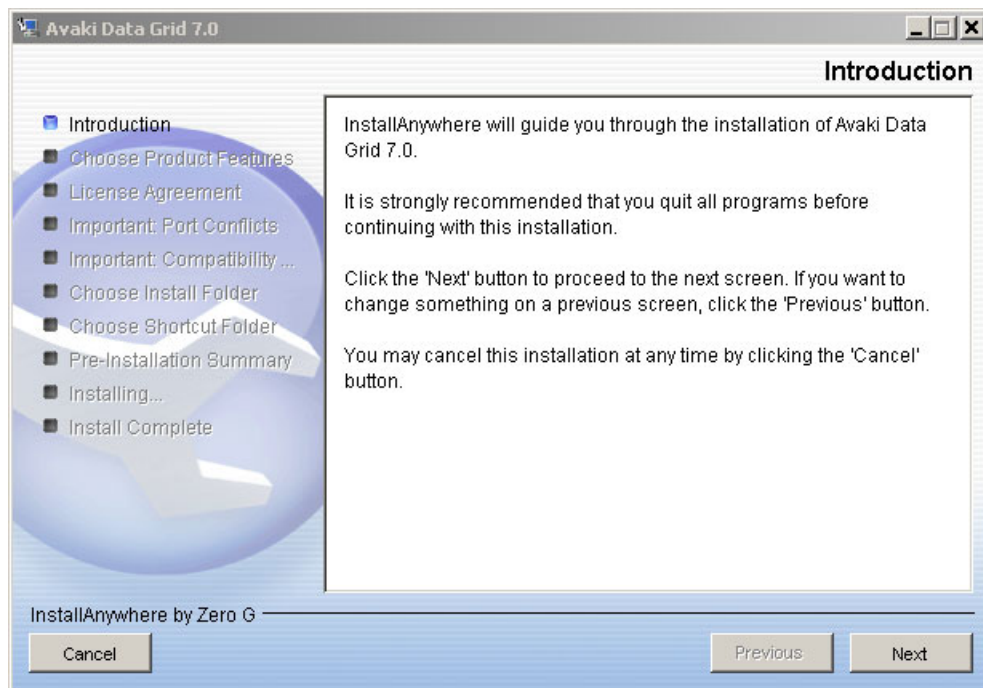
**Step 6** Click **Apply**.

You can now run the Avaki installer normally. Follow the “[Windows installation steps](#),” below, starting with [Step 2](#).

## Windows installation steps

On a machine running Windows 2003 Server or Windows XP Server or Professional, follow these steps to perform any type of Avaki software installation, including a grid domain controller, grid server, data grid access server, share server, firewall proxy server, command client, or Avaki Studio. For a machine running Windows 2003, follow the “[Preinstallation steps for Windows 2003](#),” above, before you follow the steps below.

- Step 1** Copy the Avaki installer, `install.exe`, from the CD or the web to a location on the hard disk of the Windows machine. This should be on a local disk.
- Step 2** Open `install.exe` to start the installer. The installer unpacks itself, then displays a series of interactive screens that walk you through the installation steps.
- Step 3** At the Introduction screen, click **Next**.

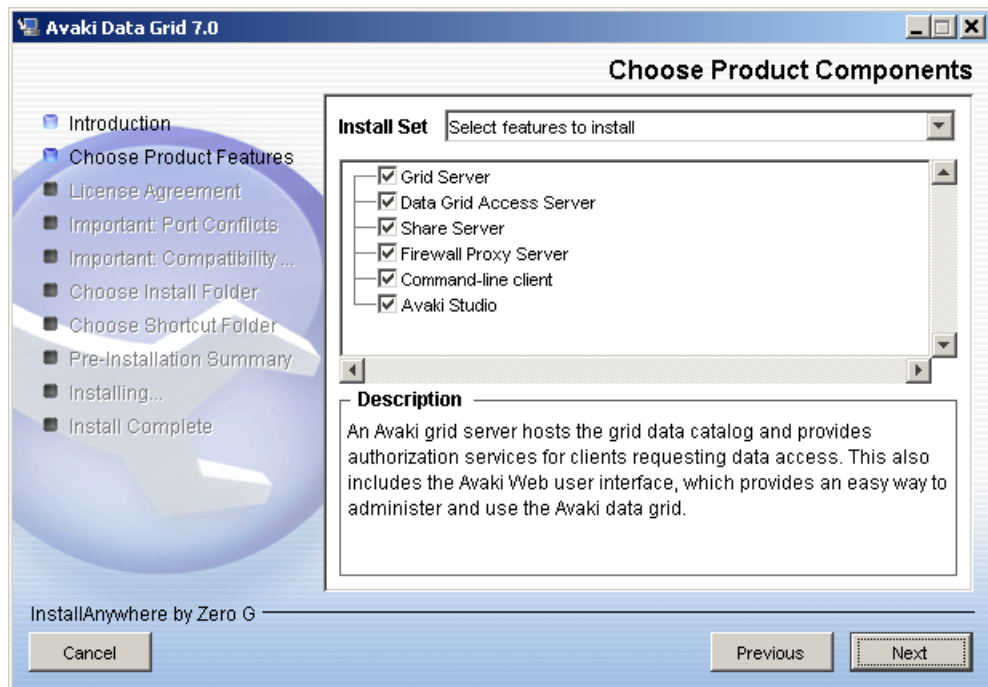


- Step 4** The installer lists the Avaki components you can install:
- **Grid Server:** A workhorse server that maintains a portion of the Avaki domain’s namespace, runs grid services such as shares and searches, and allows you to execute Avaki commands. A grid server can also be configured as a GDC.
  - **Data Grid Access Server:** Enables access to the Avaki domain via NFS clients.
  - **Share Server:** Dedicated support for linking data into the Avaki domain.

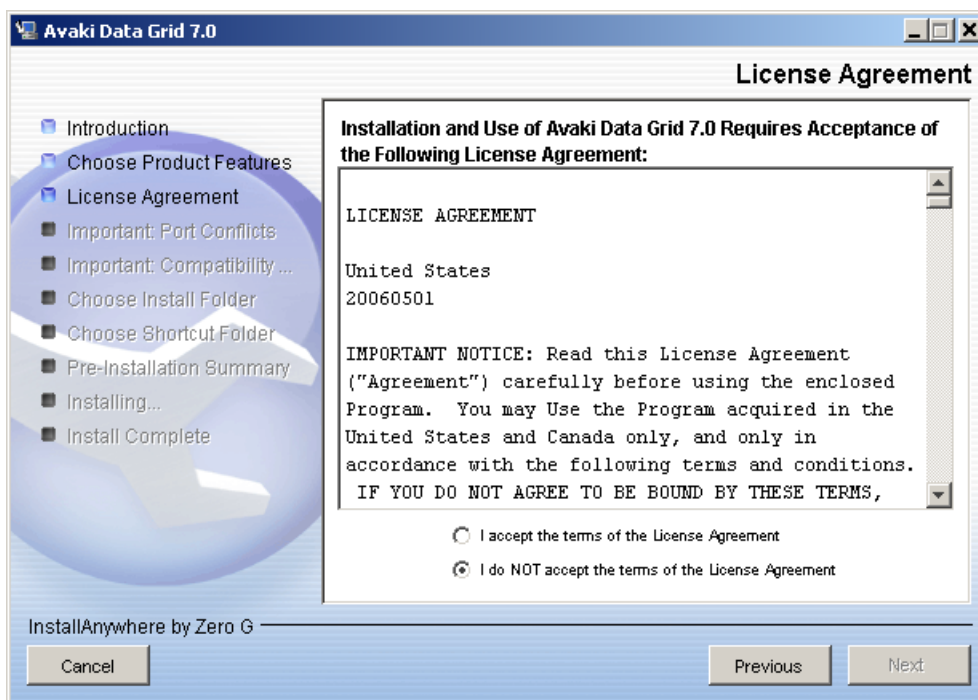
- Firewall Proxy Server: Supports multisite communications across firewalls.
- Command-line client: Command line tools only. These are primarily for administrators and advanced users.
- Avaki Studio: A graphical tool for creating Avaki data services by dragging and dropping elements into a data flow.

Click boxes to *unselect* any components you do *not* want to install, then click **Next**.

**Note** By default, all components are selected for installation. Be sure to eliminate those that you do not need on the current machine.



**Step 5** Select “I accept the terms of the License Agreement,” then click **Next**.



The installer displays the Port Conflicts screen.

**Step 6** Read the screen and click **Next** to proceed.

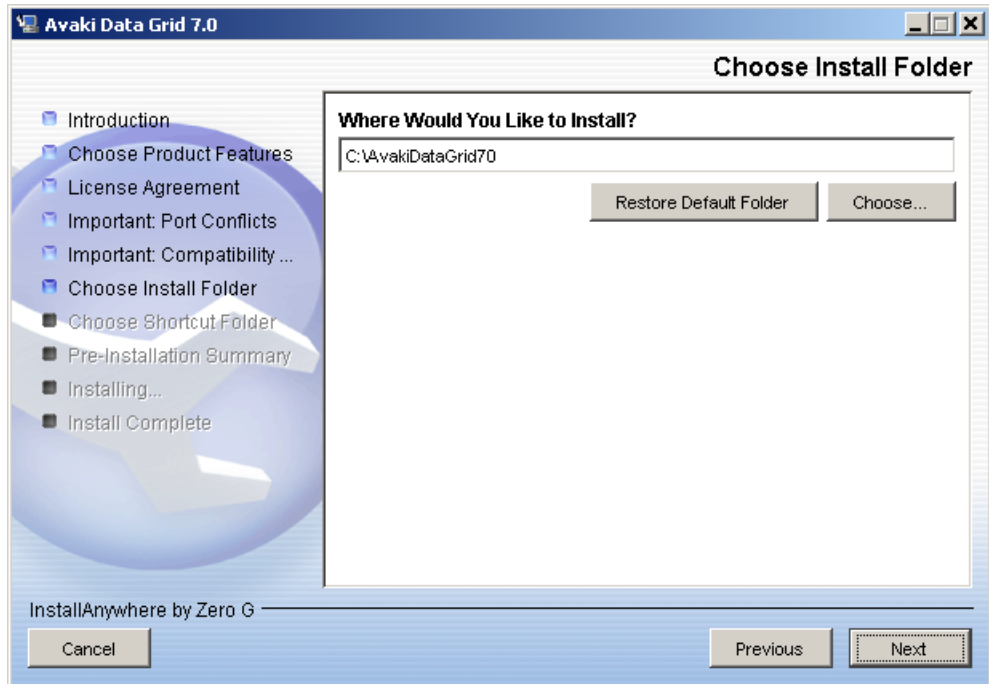
**Step 7** If you’re installing on Windows 2003, the installer displays a screen with instructions on enabling Windows 2000 compatibility mode. If you have not enabled Windows 2000 compatibility mode, click **Cancel** to quit the installer and see [“Preinstallation steps for Windows 2003” on page 22](#) for the instructions. When you’ve enabled Windows 2000 compatibility mode for the installer, restart the installer and repeat the preceding steps. Click **Next** when you reach the compatibility screen.

The installer displays the Choose Install Folder screen.

**Step 8** By default, the installer loads Avaki software into the folder `<system_drive_root>\AvakiDataGrid<release-number>` (for example, `C:\AvakiDataGrid70`). Specify a different location if desired and click **Next**.

**Caution** If you are upgrading to a new release of Avaki software, install the new release in a new folder. (Accepting the installer’s default folder is a good choice.) If you install the new release in the same folder as the older release, the upgrade will fail. See [Appendix C, “Upgrading to Avaki 7.0”](#) for instructions on performing an upgrade.

**Note** Do not install Avaki software into a folder whose pathname includes spaces. (For example, do not use the Program Files folder.)



- Step 9** The installer creates the following shortcuts (depending on which components you chose to install in [Step 4](#)):
- **DGAS:**
    - Register Data Grid Access Server as Windows service: Starts a DGAS and registers it as a service with Windows so that it will restart automatically whenever the machine reboots.
    - Unregister Data Grid Access Server as Windows service: Stops a DGAS and removes its registration as a Windows service.
  - **Documentation:** Contains links to the books in the Avaki documentation set.
  - **Grid Server:**
    - Register Grid Server as Windows service: Starts a grid server or a grid domain controller and registers it as a service with Windows so that it will restart automatically whenever the machine reboots.
    - Unregister Grid Server as Windows service: Stops a grid server or a grid domain controller and removes its registration as a Windows service.
  - **Proxy Server:**
    - Register Proxy Server as Windows service: Starts a proxy server and registers it as a service with Windows so that it will restart automatically whenever the machine reboots.
    - Unregister Proxy Server as Windows service: Stops a proxy server and removes its registration as a Windows service.
  - **Share Server:**
    - Register Share Server as Windows service: Starts a share server and registers it as a service with Windows so that it will restart automatically whenever the machine reboots.
    - Unregister Share Server as Windows service: Stops a share server and removes its registration as a Windows service.
  - **Avaki Studio:** Starts Avaki Studio.
  - **Uninstall Avaki Data Grid:** Removes Avaki software from this machine.

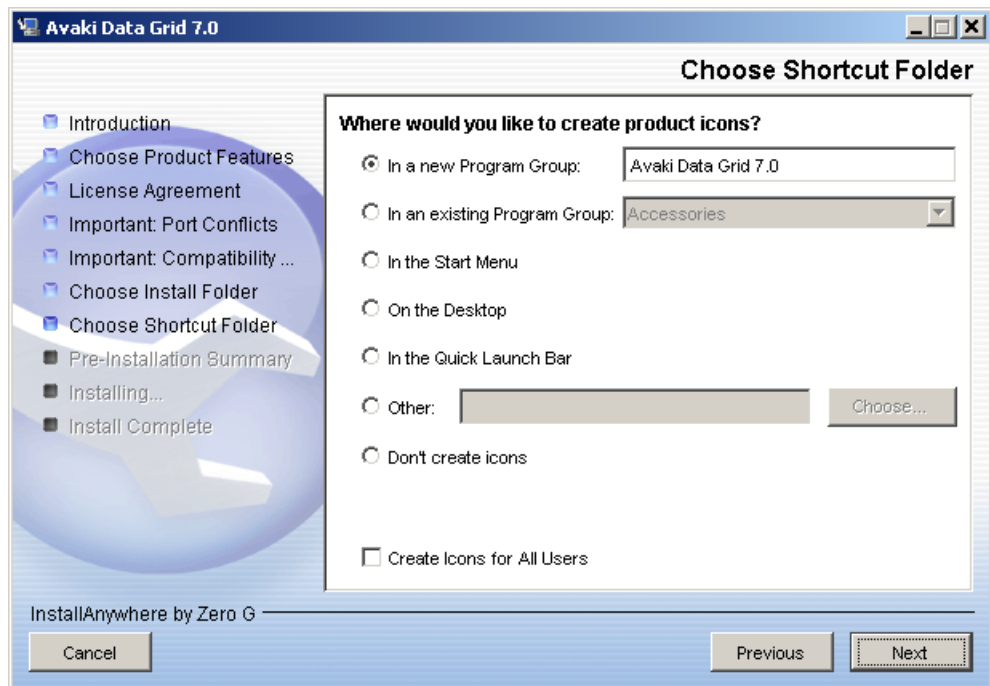
By default, the installer adds the shortcuts to a Windows program group called Avaki Data Grid 7.0. You can choose one of the following alternate locations:

- **In a new Program Group:** Specify a name for the shortcut folder.



- In an existing Program Group: Select the program group to which the Avaki shortcuts will be added.
- In the Start Menu: The Avaki Data Grid 7.0 program group will be added to the Windows Start Menu.
- On the Desktop: The Avaki Data Grid 7.0 program group will be added to the Windows desktop.
- In the Quick Launch Bar: The Avaki Data Grid 7.0 program group will be added to the Quick Launch Bar. (The Quick Launch Bar is a portion of the Windows task bar; it allows you to start programs from the desktop.)
- Other: Specify the location for the Avaki Data Grid 7.0 program group.
- Don't create icons: Select this option if you don't want the installer to create shortcuts. However, if you're installing Avaki Studio, we recommend against selecting this option. Without a shortcut, you would have to use the command line to start Avaki Studio.

Check “Create Icons for All Users” to create icons (in the location you specified) not only for your own user account but for all user accounts on this machine.

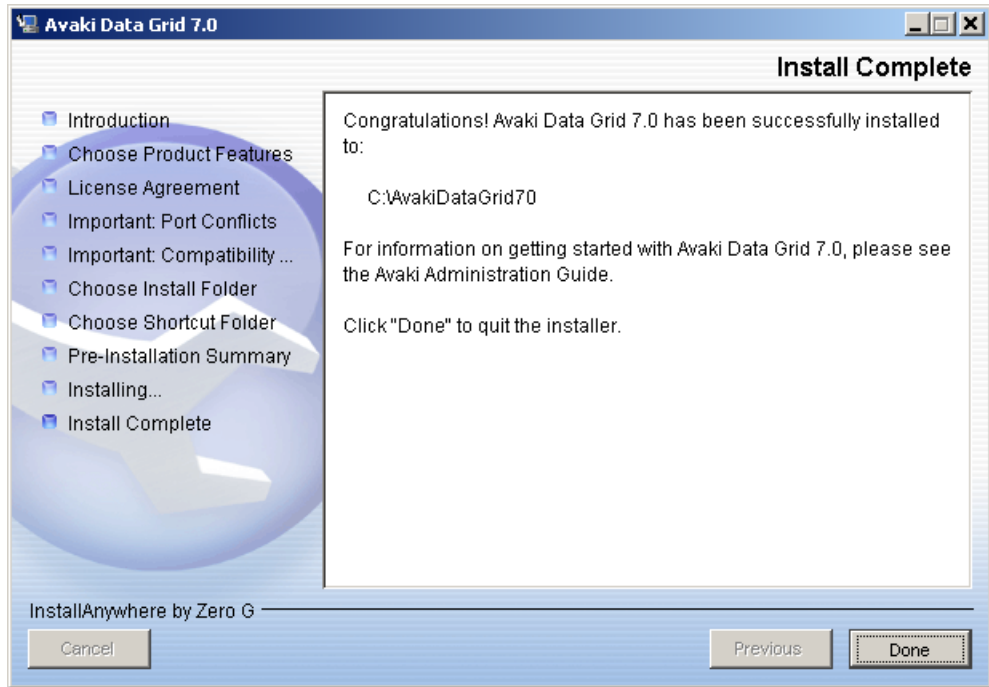


- Step 10** Review your installation choices on the Pre-Installation Summary screen. (Notice that this screen includes information on required and available disk space—scroll down to see it.) To change a choice, click Previous until you reach the appropriate screen. When you're satisfied with the choices on the Pre-Installation Summary screen, click **Install**.



Installation takes several minutes.

**Step 11** To complete the installation process, click **Done**.



## How to proceed

Perform these tasks following either a Windows or a Unix installation:

**Step 1** If you didn't do so before installing, check for port conflicts between the Avaki servers you installed and any other applications installed on the same machine. Check for port conflicts *before* you start any Avaki servers. Once a server has been joined to a grid, it's not possible to change its ports; if you discover a port conflict at that point, you'll have to re-install the server.

The list of default ports is in [“Planning use of network ports” on page 8](#). Instructions for reassigning ports in case of conflict appear in the server configuration procedures:

- Grid servers (including GDCs): [“Setting up grid servers” on page 48](#)
- Share servers: [“Setting up share servers” on page 54](#)
- DGASes: [“Setting up data grid access servers” on page 62](#)
- Proxy servers: [“Configuring a proxy server” on page 300](#)

- Step 2** Review the information in “[Setting a server’s host name or IP address](#),” below, to determine whether you need to set the host name for any Avaki servers in your grid domain.
- Step 3** Generate an SSL certificate if your site has chosen to do so (“[Generating self-signed SSL certificates](#)” on page 34).
- Step 4** Start a grid domain controller, install an SSL certificate, and configure the GDC (begin with “[Starting a primary GDC](#)” on page 37).
- Step 5** Repeat the installation procedure on the machines that will host the secondary GDC, servers, clients, and Avaki Studio. After installing those items, proceed as follows:
- The procedure for setting up a secondary GDC is in “[Starting a secondary GDC](#)” on page 43.
  - Procedures for starting and configuring grid servers, proxy servers, share servers, data grid access servers, and command clients can be found in [Chapter 3](#), “[Setting up Avaki servers and clients](#)” on page 47.
  - The procedure for starting and setting up Avaki Studio is in *Data Integration with Sybase Avaki Studio*.

## Setting a server’s host name or IP address

An Avaki GDC, grid server, DGAS, proxy server, or share server advertises its host name or IP address so that clients can communicate with it. By default the server learns its DNS name or IP address from the machine it runs on, but in any of the following situations, you might want to designate a particular IP address or DNS name for your Avaki server to advertise:

- If the server machine has more than one DNS name or more than one IP address
- If the server operates in a DHCP environment where IP addresses change, but host names are constant.
- If the server operates in a testbed environment where domain names change, but IP addresses are constant.

In any of these situations you might see an error similar to this:

```
Unable to communicate with <host-name-or-IP-address>. Host
not found. Please make sure server is running.
```

You can specify a DNS name or IP address for your Avaki server to advertise by setting a Java system property called `java.rmi.server.hostname` on the server in question. You

can set it to a hostname (host.yourdomain.com, for example) or to an IP address (172.27.3.11, for example). This process is described in “[Setting the host name property](#),” below.

## Host name changes

If an Avaki server’s host name changes, in many cases no action is required, and the server picks up the change the next time it restarts. If the server is the GDC in a multi-server Avaki domain, however, do not change the host name—if you do, the other servers in the domain won’t be able to communicate with the GDC.

On a GDC in a multiserver domain, it’s best to set `java.rmi.server.hostname` before you start the GDC for the first time, because changing the name of a running GDC can cause internal grid communications involving objects on the GDC to fail.

However, if the GDC machine’s notion of what its name is changes (which might happen if the network infrastructure changes, for example, or if you move the machine to a different network), and if you haven’t previously set `java.rmi.server.hostname`, you can use `java.rmi.server.hostname` to set the host name back to its original value.

## Setting the host name property

Follow these steps to set the property:

- Step 1** In a text editor, open the `system.properties` file for the server whose advertised host name or IP address you want to set. The path to `system.properties` varies by server type:

```
<Avaki-install-dir>/jboss/server/grid-server/conf/system.properties
<Avaki-install-dir>/jboss/server/proxy-server/conf/system.properties
<Avaki-install-dir>/jboss/server/share-server/conf/system.properties
<Avaki-install-dir>/jboss/server/ui-server/conf/system.properties
<Avaki-install-dir>/DGAS/system.properties
```

- Step 2** There is a commented-out line near the top of the `system.properties` file that looks like this:

```
java.rmi.server.hostname = myhost.mydomain.com
```

Uncomment this line by deleting the # (octothorpe), then replace the text “myhost.mydomain.com” (or whatever comes after the equal sign) with the hostname or IP address you want this Avaki server to use.

- Step 3** Save the file.

## Generating self-signed SSL certificates

During the planning phase of your Avaki installation, you or someone at your site decided how to handle SSL certificates. (See [“Planning for SSL certificates” on page 10.](#))

If you want to generate your own SSL certificates, you must execute the procedure in this section on each machine that will host an Avaki grid server (including a GDC), share server, or proxy server. (Data grid access servers do not require SSL certificates.)

During GDC or server configuration, you’ll be alerted about the current SSL certificate. You can take one of two actions when you see the security alert (see [page 40](#)):

- Choose to trust and install the current certificate. (If you followed the procedure below, the current certificate is your self-signed certificate; if not, the current certificate is Avaki’s self-signed certificate, which ships with the software.)
- Install a certificate that you have purchased from a root certification authority such as VeriSign, Inc. or Thawte, Inc.

A self-signed certificate is one for which the issuer (signer) is the same as the subject (the entity whose public key is being authenticated by the certificate). To generate your own self-signed SSL certificate, do the following on each machine that will host an Avaki server (except DGAS machines):

**Step 1** Issue the following commands to set the JAVA\_HOME and PATH environment variables:

- In Unix, in a bash shell:

```
export JAVA_HOME=<Avaki-install-dir>/jre
export PATH=<Avaki-install-dir>/jre/bin:$PATH
```

- In Windows:

```
set JAVA_HOME="<Avaki-install-dir>\jre"
set PATH="<Avaki-install-dir>\jre\bin";%PATH%
```

**Step 2** Change to the Avaki installation directory. For example, you might enter:

```
cd \AvakiDataGrid70
```

**Step 3** Run the JavaSoft JDK **keytool** utility to create a keystore in which the SSL certificate will be stored. The syntax is as follows:

```
keytool -genkey -alias <keystore-alias> -keypass
<key-pair-password> -storepass <keystore-password> -keyalg
<algorithm> -dname "CN=<common-name>,
OU=<organizational-unit>, O=<organization>, L=<locality>,
ST=<state or province>, C=<country>" -keysize <key-size>
-validity <expiration-period> -keystore
<keystore-path/filename>
```

The parameters are as follows:

| Option    | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|-------------------------------------------------------------------------------------------------|----|-------------------|---|--------------|---|----------|----|-------------------|---|---------|
| -alias    | Assigns an identity to a keystore entry. You must specify <code>rmi+ssl</code> for the alias.                                                                                                                                                                                                                                                                                                                                                                                                                                                           |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |
| -dname    | Assigns an X.500 Distinguished Name to a keystore entry. The Distinguished Name can consist of the following: <table border="1" data-bbox="471 755 1230 1137"> <tbody> <tr> <td>CN</td> <td>Common name; you must specify the host name of the grid server that the SSL certificate is for.</td> </tr> <tr> <td>OU</td> <td>Organization unit</td> </tr> <tr> <td>O</td> <td>Organization</td> </tr> <tr> <td>L</td> <td>Locality</td> </tr> <tr> <td>ST</td> <td>State or province</td> </tr> <tr> <td>C</td> <td>Country</td> </tr> </tbody> </table> | CN | Common name; you must specify the host name of the grid server that the SSL certificate is for. | OU | Organization unit | O | Organization | L | Locality | ST | State or province | C | Country |
| CN        | Common name; you must specify the host name of the grid server that the SSL certificate is for.                                                                                                                                                                                                                                                                                                                                                                                                                                                         |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |
| OU        | Organization unit                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |
| O         | Organization                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |
| L         | Locality                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |
| ST        | State or province                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |
| C         | Country                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |
| -genkey   | Creates a keystore if none exists.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |
| -keyalg   | Signifies the algorithm to be used for key-pair creation. The valid values are DSA (the default) or RSA.                                                                                                                                                                                                                                                                                                                                                                                                                                                |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |
| -keypass  | Assigns a password to a key pair. You must specify <code>rmi+ssl</code> for the password.                                                                                                                                                                                                                                                                                                                                                                                                                                                               |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |
| -keysize  | Specifies a key size. The valid range is 512 to 1,024 bits; the default is 1,024 bits.                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |
| -keystore | Specifies the name and location of a keystore.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |    |                                                                                                 |    |                   |   |              |   |          |    |                   |   |         |

| Option     | Description                                                                               |
|------------|-------------------------------------------------------------------------------------------|
| -storepass | Assigns a password to a keystore. You must specify <code>rmi+ssl</code> for the password. |
| -validity  | Sets an expiration period. The default is 90 days.                                        |

For example:

```
keytool -genkey -alias rmi+ssl -keypass rmi+ssl -storepass
rmi+ssl -keyalg RSA -dname "CN=bedrockgdc.slate.com,
OU=Slate, O=Slate, L=Bedrock, ST=MA, C=US" -keysize 1024
-validity 9490 -keystore /opt/grid/AvakiDataGrid/
jboss/server/grid-server/conf/.keystore-new
```

**Step 4** To replace the keystore certificate file that is distributed with Avaki with your new self-signed certificate, issue the following commands:

- In Unix:

```
cp /<Avaki-install-dir>/jboss/server/<server-type>/conf/
.keystore /<Avaki-install-dir>/jboss/server/<server-type>/
conf/.keystore-old
```

```
cp <Avaki-install-dir>/jboss/server/<server-type>/conf/
.keystore-new <Avaki-install-dir>/jboss/server/
<server-type>/conf/.keystore
```

where `<server-type>` is `grid-server`, `share-server`, or `proxy-server`. For example:

```
cp /opt/grid/AvakiDataGrid/jboss/server/grid-server/conf/
.keystore /opt/grid/BedrockGrid/jboss/server/grid-server/
conf/.keystore-old
```

```
cp /opt/grid/BedrockGrid/jboss/server/grid-server/conf/
.keystore-new /opt/grid/AvakiDataGrid/jboss/server/
grid-server/conf/.keystore
```

- In Windows:

```
copy <Avaki-install-dir>\jboss\server\<server-type>\conf\
.keystore <Avaki-install-dir>\jboss\server\<server-type>\
conf\.keystore-old
```

```
copy <Avaki-install-dir>\jboss\server\<server-type>\conf\
.keystore-new <Avaki-install-dir>\jboss\server\
<server-type>\conf\.keystore
```



where <server-type> is grid-server, share-server, or proxy-server. For example:

```
copy "c:\AvakiDataGrid\jboss\server\grid-server\
conf\.keystore" "c:\AvakiDataGrid\jboss\server\
grid-server\conf\.keystore-old"
```

```
copy "c:\AvakiDataGrid\jboss\server\grid-server\
conf\.keystore-new" "c:\AvakiDataGrid\jboss\server\
grid-server\conf\.keystore"
```

Start the grid domain controller (see [“Starting a primary GDC,”](#) below) or server (see [Chapter 3, “Setting up Avaki servers and clients”](#)). After the GDC or server starts, you can install the self-signed certificate. Certificate installation is described in the section [“Configuring a primary GDC”](#) on page 41.

## Starting a primary GDC

### Using nondefault ports

If you want to run your GDC on ports other than the default ports listed in [“Planning use of network ports”](#) on page 8, see [“Configuring a grid server to use nondefault ports”](#) on page 50 for instructions.

### Auto-restart or not?

You can start your grid domain controller (or any other Avaki server) with or without the automatic restart feature. If you use auto-restart, the GDC or server restarts itself when its machine reboots. Auto-restart is a good choice for production networks—those installations where the data grid will be in day-to-day use doing real work.

To start a GDC, choose one of these procedures:

- [“Starting the GDC without auto-restart”](#) on page 38
- [“Starting the GDC with auto-restart”](#) on page 38

These procedures cover both Windows and Unix.

After starting a GDC, proceed to [“Configuring a primary GDC”](#) on page 41.

## Starting the GDC without auto-restart

To start a GDC without auto-restart, execute the **grid-server --start** script in the Avaki installation directory.

The server takes a minute or two to come up. Startup is complete when you see a message similar to this:

```
2003-06-30 13:50:26,951 INFO
[org.jboss.system.server.Server, main] JBoss (MX
MicroKernel) [3.0.6 (CVSTag=JBoss_3_0_6 Date=200301260037)]
Started in 1m:3s:212ms
```

---

**Note** To stop the GDC, enter **Ctrl-C** in its window.

Skip to the procedure [“Configuring a primary GDC” on page 41](#).

## Starting the GDC with auto-restart

**In Unix:** To start a GDC with auto-restart, log in as root and execute the **grid-server** script in the Avaki installation directory. Its syntax is

```
grid-server --register [--user=<user>]
```

where

<user> is the nonadministrative user account that you set up to run Avaki services (see [“Configuring user accounts” on page 11](#)).

For example:

```
$ grid-server --register --user=avaki
```

(If you omit the --user option, you’ll be prompted to enter a user ID.)

Skip to the procedure [“Configuring a primary GDC” on page 41](#).

**In Windows:** To start a GDC with auto-restart, do either of the following:

- Open the Start menu. In the Avaki Data Grid 7.0 program group, select **Grid Server**, then select the **Register Grid Server as Windows service** shortcut.
- In the Avaki installation directory (which defaults to <system\_drive\_root>\AvakiDataGrid<release-number> (for example, C:\AvakiDataGrid70), enter this command:

```
C:\AvakiDataGrid70> grid-server --register
```

The server prompts you for:

- A user name. Enter the name of the nonadministrative user account with logon-as-service privileges that you set up to run Avaki services (see [“Configuring user accounts” on page 11](#)). Enter the Windows domain name before the user name, for example, BEDROCK\avaki.
- A password. Enter the password for the user you entered above.

The server takes a minute or two to start. When it’s done, you can find an entry for the grid server in the machine’s Services list (Start > Control Panel > Administrative Tools > Services).

---

**Note** Use the Windows Services list to stop and restart Avaki servers that are registered as services. Right-click on the server name (for example, Avaki-Grid-Service-1096608202) and select **stop** or **restart** from the menu.

## Installing an SSL certificate

When you access the web UI for a newly started Avaki server (including a GDC), you’ll see a security alert (shown in [Step 1](#) below) that pertains to this server’s current SSL certificate. The current SSL certificate is either the certificate that shipped with Avaki software, or your self-signed certificate if you chose to generate one (see [“Generating self-signed SSL certificates” on page 34](#)). This procedure shows how to install the current certificate.

If you want to buy a certificate from a certification authority rather than installing Avaki’s certificate, follow the CA’s instructions for installing their certificate.

Once you’ve started a grid domain controller (or other Avaki server), follow these steps to install the current SSL certificate:

**Step 1** Point your web browser to this address:

```
http://<grid-server-name>:7080
```

A window similar to this appears:



**Step 2** Click **View Certificate**.

**Step 3** On the General tab, click **Install Certificate...**

**Step 4** The Certificate Import Wizard displays its welcome message; click **Next**.

**Step 5** Specify a location for your certificate store, or allow the OS to select a location. Click **Next**.

**Step 6** Click **Finish**.

**Step 7** A dialog box asks if you want to add the certificate to the store. Click **Yes**.

**Step 8** Click **OK**.

**Step 9** Back in the Security Alert window, click **Yes** in response to “Do you want to proceed?”

When you have finished with the security alert for the SSL certificate, proceed directly to the next section, “[Configuring a primary GDC.](#)”

## Configuring a primary GDC

Before starting this procedure, you must start the primary GDC (“[Starting a primary GDC](#)” on page 37) and install an SSL certificate (“[Installing an SSL certificate](#)” on page 39).

- Step 1** When your browser connects to <http://localhost:7080>, the Create Grid Domain screen appears.

The screenshot shows the Avaki Data Grid Configuration interface. At the top left is the Avaki logo, and at the top right is a 'Help' link. Below this is a dark green header with the text 'Data Grid Configuration'. The main content area is titled 'Create Grid Domain' and contains the following text: 'To create a grid domain, specify the name of the domain and the connect port of the grid domain controller.' There are two input fields: 'Grid domain name (up to 30 characters):' and 'Grid domain controller connect port:'. The second field contains the value '3099'. A 'Submit' button is located at the bottom left of the form area.

Copyright © 2002-2006 Sybase Inc. All Rights Reserved

In the Grid domain name field, enter a unique name for the Avaki grid domain you are creating. Several restrictions apply to domain names:

- Do not use the name “dataservice” (nor variant capitalizations: “DataService,” “DATASERVICE,” etc.); it conflicts with a default used by Avaki software.
  - A domain name can have a maximum of 30 characters.
  - You can use letters, digits, underscores (\_) and dots (.).
  - Do not use spaces or other special characters.
- Step 2** If you started the GDC with a port other than the default connect port, 3099, enter that other port number in the Grid domain controller port field. (See the [Glossary on page 349](#) for more information on [connect ports](#).) If this step is necessary in your case, a Sybase customer service representative will advise you.
- Step 3** Click **Submit**.
- The system creates a new Avaki domain, which takes a minute or two. Then the login screen appears.
- Step 4** Log in as Administrator; the password is Administrator. DefaultAuthService is the only authentication service for a newly created Avaki domain. Click **Sign In**.

The Welcome screen appears. You can return to this main menu from anywhere in the Avaki web interface by clicking the **Home** link at the top of the page.

- Step 5** Change the Administrator password immediately. To do so, navigate to the Select User screen:

User Management > View and modify users

**AVAKI** Welcome, Administrator • [Logout](#) • [Help](#)

**Data Grid User Management**

[Home](#) • [Manage Users & Groups](#) • Select Users

Current Domain: Burlington Current Server: Antimony.avaki.local

### Select User

To modify a user's personal information, download a user interconnect ID, or change a password, select the user and click the **View/Edit** button. To delete a user from a grid domain, select the user and click **Delete All Checked**. To configure the user ID (UID) that is mapped to a user account, select the user and access the [Update UIDs](#) screen.

If you have a two-way connection with a remote domain, you may want to enable the administrator of the remote domain to view users in your domain and add them to groups in the remote domain. To do this, you must give the administrator permission to view and edit those users, as follows:

1. Obtain the interconnect ID that represents the administrator of the remote domain.
2. In the table below, select the users whose settings you want to allow the remote administrator to modify.
3. Go to the [Expose Users](#) screen to continue.

**Note:** The remote administrator will be able to edit only the users you selected, but the administrator will be able to view *all* the users in your domain.

[All users](#)

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

| Select                   | Username      | Authentication Service | View/Edit User            | Attributes                 | Security                 |
|--------------------------|---------------|------------------------|---------------------------|----------------------------|--------------------------|
| <input type="checkbox"/> | Administrator | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | MessagingUser | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

Total number of users in domain: 2

- Step 6** Click **View/Edit** for the Administrator account.

- Step 7** A new screen appears. In the Change Password area at the bottom, enter the new password in both fields.

**Change Password**

To change the user's password, enter the new password in the "New password" and "Confirm new password" fields.

New password:

Confirm new password:

- Step 8** Click **Submit** to save the new password.
- Step 9** If this GDC will host any Avaki database connectors, you must install JDBC drivers for the databases that will be used. For instructions, see [“Installing database JDBC drivers” on page 49](#). When you’re done, be sure to come back and complete the instructions in this chapter.

## Starting a secondary GDC

To increase your Avaki domain’s availability and fault tolerance, you can set up a secondary grid domain controller. The secondary GDC is a hot standby that handles requests if the primary GDC is unreachable.

Set up the secondary GDC on a machine that meets the GDC requirements outlined on [page 15](#). We recommend that you configure only one secondary GDC per domain.

If you don’t want to set up a secondary GDC, skip to the section [“Setting up administrative accounts in Avaki” on page 45](#).

---

**Note** If you want to have a secondary GDC in your Avaki domain, you must set it up after you set up the primary GDC, but before you set up the other Avaki servers in the domain. If you set up other servers before the secondary GDC, the Avaki failover mechanism cannot function properly.

To start a secondary GDC, follow these steps.

- Step 1** Install Avaki software on the machine that will serve as your secondary GDC. See [“Installing in Unix” on page 16](#) or [“Installing in Windows” on page 22](#).
- Step 2** If you have chosen to generate self-signed SSL certificates for your Avaki servers, set up the secondary GDC machine to do so. For instructions, see [“Generating self-signed SSL certificates” on page 34](#).

- Step 3** Start a grid server on the secondary machine, as described in the section [“Starting a primary GDC” on page 37](#). (Do not go on to configure the secondary GDC; return to this procedure instead.)
- Step 4** Install an SSL certificate on the secondary GDC machine. For instructions, see [“Installing an SSL certificate” on page 39](#).
- Step 5** Log in to the grid domain as Administrator.
- Step 6** Navigate to the Create Secondary Grid Controller screen:

Home > Server management > Create secondary GDC

**Create Secondary Grid Domain Controller**

A secondary grid domain controller (GDC) provides failover capabilities for the primary GDC. To initialize a secondary GDC, specify its IP or host name and connect port.

DNS name or IP address of secondary GDC:

Connect port of secondary GDC:

- Step 7** In the Secondary GDC machine field, enter the host name or IP address of the machine that you’re setting up as the secondary GDC.
- Step 8** If you started the secondary GDC on a port other than the default port, 3099, enter that other port number in the Connect port of secondary GDC field.
- Step 9** Click **Submit**.

By default, the secondary GDC receives updates from the primary every 15 minutes. (Updates include changes to user accounts and groups.) To change the update interval or force an immediate update, use the Configure Secondary GDC Refresh Interval screen (Home > Server management > Refresh secondary grid domain controller).

- Step 10** If this GDC will host any Avaki database connectors, you must install JDBC drivers for the databases that will be used. For instructions, see [“Installing database JDBC drivers” on page 49](#). When you’re done, be sure to come back and complete the instructions in this chapter.

For information on failover from the primary GDC to the secondary, see [“Managing failover” on page 111](#).



---

# Setting up administrative accounts in Avaki

When you have installed Avaki software and started the GDC for your Avaki domain, you must set up administrative accounts. This section discusses the groups to which Avaki administrators belong and outlines recommended steps for setting up administrators for the Avaki domain.

## Administrative groups

A new Avaki domain has two default groups for users who perform administrative functions:

- **UserAdministrators**

Members of the UserAdministrators group can perform tasks related to user accounts and groups: creating, importing, or deleting accounts or groups; adding users to groups and removing users from groups; changing passwords; setting permissions. The only user-related function that UserAdministrators cannot perform is adding members to the Administrators group.

By default, the UserAdministrators group has no members. Members of the Administrators group (and members of UserAdministrators, once there are some) can add users as needed.

- **Administrators**

Members of the Administrators group can perform all administrative tasks, including user administration (see under UserAdministrators, above, for details). You must be a member of the Administrators group to add a user to the Administrators group, to change the ownership of any grid object that belongs to a member of the Administrators group, to configure Avaki servers, and to integrate and delete external authentication services. Members of the Administrators group are not subject to the permissions in Avaki access control lists; they can read, write, delete, and change the ownership and permissions of any Avaki object in their grid domain. Putting a member of the Administrators group on an ACL deny list has no effect.

In a newly installed Avaki domain, the Administrators group has one member, Administrator, who can add other users as needed.

For descriptions of the other default grid groups, see the *Sybase Avaki EII Overture*.

## Overview: Setting up Avaki administrators

To set up new administrators (in addition to the default Administrator account) in a new Avaki domain, do the following:

1. Determine who the administrative users will be and which administrative group (Administrators or UserAdministrators) each administrator will belong to.
2. Create or import user accounts for each administrator. For instructions on creating a grid user account, see [“Creating grid user accounts” on page 168](#).

Before you can import any user accounts from an external directory service, you must integrate the directory service into your Avaki domain; see [“Managing authentication services” on page 148](#). Then, for instructions on importing a user account from an authentication service, see [“Importing user accounts from an LDAP authentication service” on page 157](#), or [“Importing user accounts from an NIS authentication service” on page 163](#).

3. Assign each account to the appropriate administrative group (Administrators or UserAdministrators). See [“Adding users to groups” on page 192](#).

---

**Note** Avaki administrators need to understand how access control works in Avaki grids. We recommend that administrators become very familiar with the chapter on authentication and access control in the *Sybase Avaki EII Overture*.

---

## How to proceed

When you have completed the installation procedures described in this chapter, you’re ready to configure Avaki software. Follow the instructions in [Chapter 3, “Setting up Avaki servers and clients”](#).

# *Setting up Avaki servers and clients*

---

In this chapter, we explain how to configure the servers and clients that make up a data grid. (For an overview of the servers and clients, see the *Sybase Avaki EII Overture*.)

This chapter covers the following topics:

- “Server and client requirements” on page 48
- “Setting up grid servers” on page 48
- “Setting up share servers” on page 54
- “Setting up data grid access servers” on page 62
- “Setting up NFS clients” on page 92
- “Setting up CIFS clients” on page 94
- “Setting up proxy servers and routing tables” on page 94
- “Setting up command clients” on page 95

---

## Server and client requirements

A machine that participates in a grid domain can be either a server or a client. A server provides disk space and services to the grid domain; a client can access the grid domain but does not provide disk space or services.

For every server or client (except NFS and CIFS clients) that you want to add to a grid domain, you must have:

- A machine for the server or client that meets the requirements in [“Loading and starting Avaki Data Grid software” on page 15](#);
- Access to the server or client machine through a local user account; and
- An Avaki distribution that is installed on the server or client machine. For installation instructions, see [Chapter 2, “Installing Avaki software”](#).

---

## Setting up grid servers

A grid server makes the local machine’s storage and computing resources available to the Avaki domain. Once a grid server is configured on a machine, you can selectively share the machine’s files and directories into the data catalog, connect share servers on other machines, configure caching, and execute data services and queries against the virtual database.

### Installing Avaki software

Before setting up a grid server, do the following:

- Step 1** Start and configure a grid domain controller (GDC) for this grid domain. See [“Installing Avaki software” on page 13](#) for instructions.
- Step 2** We recommend making two user accounts (either local accounts or directory service accounts) available from the grid server machine: one that has administrative privileges, and one that does not. (In Unix, a user who has administrative privileges logs in as root; in Windows, a user who has administrative privileges is listed in the Administrators group.) A user account with administrative privileges is required in order to run certain Avaki commands, but for security reasons, an account with administrative privileges should not be used to run most Avaki servers. In Windows, configure the

account that does not have administrative privileges so that the user has permission to log on as a service.

- Step 3** Run the Avaki installer on the target machine, and select a Grid Server installation. For instructions, see [“Installing in Unix” on page 16](#) or [“Installing in Windows” on page 22](#).
- Step 4** Log in to the target machine.
- Step 5** (Optional) Specify a host name or IP address for your grid server to advertise if the default (the server machine’s name) is unsuitable. For instructions, see [“Setting a server’s host name or IP address” on page 32](#).
- Step 6** If you have chosen to generate self-signed SSL certificates for your Avaki servers, set up this machine to do so. For instructions, see [“Generating self-signed SSL certificates” on page 34](#).

Proceed to the next section, [“Installing database JDBC drivers.”](#)

## Installing database JDBC drivers

You must install one or more JDBC drivers on any grid server (including GDCs) that will host Avaki database connectors. (Database connectors enable Avaki to retrieve data from databases.) You’ll need a JDBC driver for each database that will be accessed.

---

**Note** If you need two versions of the same JDBC driver on one grid server, follow the steps here to install the first version. Then refer to the *Avaki Provisioning and Advanced Data Integration Guide* for instructions on installing the second version.

Follow these steps to install JDBC drivers.

- Step 1** Obtain the JDBC drivers. See [page 6](#) for a list of supported databases and associated drivers.
- Step 2** Copy the JAR file for each driver to the local directory <Avaki-install-dir>/drivers on the grid server machine.
- Step 3** If the grid server is running, restart it:
  - If the grid server is registered as a service, use the tools provided by the operating system to stop and start the grid server.
  - If the grid server is not registered as a service, enter **Ctrl-C** in the window where the grid server is running to stop it. To restart it, enter **grid-server --start**.

To continue setting up the grid server, proceed to the next section, [“Configuring a grid server to use nondefault ports.”](#)

## Configuring a grid server to use nondefault ports

If the ports that grid servers use by default conflict with ports already in use on the grid server machine, you can specify other ports for your grid server to use. (For a list of default ports, see [“Planning use of network ports” on page 8.](#)) Follow these steps to change the default ports.

---

**Note** Do not reassign ports on an Avaki server once it’s been joined to an Avaki domain; doing so will cause communication problems between servers. If the server has been joined to a domain, you must re-install and reconfigure the server, change the ports, and join the new server to the Avaki domain.

- Step 1** To reassign port 15000 (the TDS port), set the `com.sybase.avaki.tdsPort` system property. See [“Setting system properties” on page 128](#) for information on changing the values of system properties and [“Setting the TDS port” on page 146](#) for information on `com.sybase.avaki.tdsPort`.
- Step 2** In a text editor, open `<Avaki-install-dir>/jboss/server/grid-server/conf/bindings.xml`. Ten ports are specified in this file: 1098, 3099, 7083, 4444, 8089, 8090, 8092, 8093, 8443, and 7080. Replace as many port numbers as necessary with values appropriate for this machine.

---

**Note** Two RMI ports are linked to the connect port, which defaults to 3099. These port numbers are determined by adding 1 and 2 to the connect port number—for example, when you use the default connect port number, 3099, the linked RMI ports are 3100 and 3101. When you select port numbers, be sure that the two ports after the connect port are also unused. (It is not possible to set the linked RMI port numbers independently of the connect port number.)

- Step 3** Record the port number you enter in place of 3099 (the second port number in the file); this is the grid server’s connect port, and you must supply the connect port number when you connect this grid server to the grid domain.
- Step 4** In a text editor, open `<Avaki-install-dir>/jboss/server/grid-server/conf/share-server.ports`. Two ports are specified in this file: 8512 and 8511. If necessary, replace these port numbers with values appropriate for this machine. (We suggest using two consecutive available ports.)

## Starting a grid server

This section explains how to start a grid server with and without the automatic restart feature, which restarts the grid server when its machine reboots.

---

**Note** If this grid server will support Avaki shares and you want those shares to have write access to their underlying local directories, the user account under which the grid server runs must have write privileges to the local directories in question. For more information, see [“Configuring user accounts” on page 11](#).

### Starting the grid server without auto-restart

Execute the **grid-server --start** script in the Avaki installation directory. (When you don't use auto-restart, the grid server runs under the OS user account you used to log in and start it. Any Avaki shares associated with this grid server will have the same privileges as the OS user account that started the grid server.)

The server takes a minute or two to come up. Startup is complete when you see a message similar to this:

```
2003-06-30 13:50:26,951 INFO
[org.jboss.system.server.Server, main] JBoss (MX
MicroKernel) [3.0.6 (CVSTag=JBoss_3_0_6 Date=200301260037)]
Started in 1m:3s:212ms
```

---

**Note** To stop the grid server, enter **Ctrl-C** in its window.

Skip to the procedure [“Configuring a grid server” on page 52](#).

### Starting the grid server with auto-restart

**In Unix:** Log in as root and execute the **grid-server** script in the Avaki installation directory. Its syntax is:

```
grid-server --register [--user=<user>]
```

where

<user> is a user account that you set up to run Avaki services (see [“Configuring user accounts” on page 11](#)).

For example:

```
$ grid-server --register --user=avaki
```

(If you omit the `--user` option, you'll be prompted to enter a user ID.)

Skip to the procedure [“Configuring a grid server,”](#) below.

**In Windows:** Do either of the following:

- Open the Start menu. In the Avaki Data Grid program group, select **Grid Server**, then select the **Register Grid Server as Windows service** shortcut.
- In the Avaki installation directory (which defaults to `<system_drive_root>\AvakiDataGrid<release-number>` (for example, `C:\AvakiDataGrid70`)), enter this command:

```
C:\> grid-server --register
```

The server prompts you for:

- A user name. Enter the name of the user account with logon-as-service privileges that you set up to run Avaki services (see [“Configuring user accounts”](#) on page 11). Enter the Windows domain name before the user name, for example, `BED-ROCK\avaki`.
- A password. Enter the password for the user you entered above.

The server takes a minute or two to start. When it's done, you can find an entry for the grid server in the machine's Services list (Start > Control Panel > Administrative Tools > Services).

---

**Note** Use the Windows Services list to stop and restart Avaki servers that are registered as services. Right-click on the server name (for example, `Avaki-Grid-Service-1096608202`) and select **stop** or **restart** from the menu.

## Configuring a grid server

Follow these steps to configure a grid server that has just been started.

**Step 1** Install an SSL certificate on the grid server machine. For instructions, see [“Installing an SSL certificate”](#) on page 39.

**Step 2** On either the grid server machine or the GDC machine, point a web browser to

```
http://<gdc-name>:7080
```



where <gdc-name> is the name of the grid domain controller to which you're joining the new grid server.

---

**Note** When you connect a new server to an existing Avaki domain, the connection is made *from* the GDC *to* the server—so make sure your browser is connected to the GDC and not to the grid server machine.

**Step 3** Log in to Avaki as a member of the Administrators group.

**Step 4** Navigate to the Connect Grid Server screen:

Home > Server management > Connect new grid server

**Connect Grid Server**

To connect a grid server to a domain, install a grid server on the machine where the server will reside, start the grid server, and then specify the IP address or DNS name of the machine below, as well as the server's connect port number.

IP address or DNS name of server:

Connect port of server:

**Step 5** In the IP address or DNS name of server field, enter the fully qualified DNS name or IP address of the machine on which the grid server will run. (A fully qualified DNS name is `bedrock.sybase.com`, for example, rather than just `bedrock`.)

**Step 6** If you started the grid server on a port other than the default connect port, 3099, enter that port number in the Connect port of server field.

---

**Note** You can find an Avaki server's connect port (as well as its name) in its `connectinfo.txt` file, which is generated when the server starts. The path to the file is:

<Avaki-install-dir>/jboss/server/grid-server/log/connectinfo.txt

(<Avaki-install-dir> is the local directory where Avaki software is installed.)

**Step 7** Click **Submit** to set up the grid server.

---

## Setting up share servers

This section explains the purpose of Avaki share servers and tells you how to set up share servers in different situations:

- [“About share servers” on page 54](#) describes the purpose and function of share servers.
- [“Setting up a single share server” on page 56](#) explains how to configure a share server that will be the only stand-alone share server on its machine.
- [“Configuring multiple share servers on one machine OR configuring share servers to use nondefault ports” on page 59](#) explains how to configure two or more share servers to run on the same host.

Each share server is associated with a grid server or grid domain controller. Before performing any procedures to set up a share server, we recommend that you set up at least one grid server or GDC. (See [“Setting up grid servers” on page 48.](#))

---

**Note** Before you disconnect or remove a share server, make sure there are no Avaki shares using the share server. You can disable the shares by taking them off line (see [“Taking Avaki shares off-line” on page 285](#)) or you can reassign them to another share server or grid server.

### About share servers

A share server provides dedicated support for Avaki shares. (Shared directories, or Avaki shares, are links that allow files or directories in the local file system to appear in the grid data catalog.)

Sybase recommends that you place your share server as close as possible to its data—preferably on the same physical machine. If the share server accesses the data over an NFS mount, performance is likely to suffer.

### Benefits of share servers

Share servers can provide two kinds of benefits:

- Performance.  
Each grid server includes a share server, but if you have many Avaki shares (or even one very large Avaki share) associated with a single grid server, you can improve performance by creating one or more dedicated share servers.

- Security.
  - You can set up a share server that is able to share only files and directories owned by a particular local user.
  - You can set up a share server that provides (or prohibits) write access to the underlying local file system. The share server runs under an operating system user account; the share server’s write privileges are those of the user account. For more information, see [“Configuring user accounts” on page 11](#).

---

**Note** Before you set up a share server, consider which OS user account the share server will run under, and what privileges you will assign to the account. We recommend creating at least one account for Avaki software to use. You might want to create several such accounts.

## Multiple share servers

In some cases, it makes sense to have multiple share servers serving a single Avaki share. When data is stored in network attached storage (NAS) or a storage area network (SAN), it is possible for multiple machines to have equivalent access to the same data. In the NAS case, multiple machines can mount the file server via a native protocol such as NFS or CIFS. Similarly, some SAN configurations provide multiple machines on the network transparent access to common storage devices. Multiple machines with equivalent access to the same data can provide equivalent access to Avaki shares by running share servers. If one share server fails or becomes unresponsive, another share server takes over transparently. An Avaki share with a single share server can be a performance bottleneck and a single point of failure; multiple share servers eliminate these problems.

Prerequisites for configuring several share servers on a single Avaki share:

- Each share server machine must have an equivalent view of the data in the share.
- The local file system path to the root of the data being accessed must be the same across all share server machines.
- An Avaki share and all its share servers must be connected to the same grid server.
- You must have read and write permission on all share servers in the configuration.

For instructions on adding share servers to an Avaki share, see [“Adding share servers to Avaki shares” on page 263](#).

## Setting up a single share server

Use this procedure to set up a share server on a machine where no other share servers are running.

---

**Note** Although each grid server includes a share server, you can also use this procedure to set up a standalone share server on a machine that is also running a grid server.

- Step 1** Run the Avaki installer on the target machine and select a Share Server installation. (For installation instructions, see [“Installing in Unix” on page 16](#) or [“Installing in Windows” on page 22](#).) You can install servers of more than one type (for example, a grid server and a share server) at the same time and run them out of the same installation directory.
- Step 2** (Optional) Specify a host name or IP address for your share server to advertise if the default (the server machine’s name) is unsuitable. For instructions, see [“Setting a server’s host name or IP address” on page 32](#).
- Step 3** If you have chosen to generate self-signed SSL certificates for your Avaki servers, set up the share server to do so. For instructions, see [“Generating self-signed SSL certificates” on page 34](#).
- Step 4** If you want to limit the share server to accessing files owned by a particular OS account, log in as that user to the machine on which the share server will run.
- Step 5** To start the share server:

**In Windows:** You can start a share server with or without the auto-restart feature; both methods are described here.

- Do one of the following to start the share server with auto-restart enabled:
  - Open the Start menu. In the Avaki Data Grid program group, select **Share Server**, then select the **Register ShareServer as Windows Service** shortcut.
  - In the Avaki installation directory (which defaults to `<system_drive_root>\AvakiDataGrid<release-number>` (for example, `C:\AvakiDataGrid70`)), enter this command:

```
C:\> share-server --register
```

The server prompts you for:

- A user name. Enter the name of the nonadministrative user account with logon-as-service privileges that you set up to run Avaki services (see [“Configur-](#)

ing user accounts” on page 11). Enter the Windows domain name before the user name, for example, BEDROCK\avaki.

- A password. Enter the password for the user you entered above.

The server takes a minute or two to start. When it’s done, you can find an entry for the proxy server in the machine’s Services list (Start > Control Panel > Administrative Tools > Services).

---

**Note** Use the Windows Services list to stop and restart Avaki servers that are registered as services. Right-click on the server name (for example, Avaki-Share-Service-1096608202) and select **Stop** or **Restart** from the menu.

- To start a share server without enabling auto-restart, change to the Avaki installation directory (by default, <system\_drive\_root>\AvakiDataGrid<release-number> (for example, C:\AvakiDataGrid70)) and issue a command of this form:

```
share-server --start [--user=<user>]
```

where <user> is the nonadministrative user account that you set up to run Avaki services. (For more information, “Configuring user accounts” on page 11)

For example:

```
$ share-server --start --user=avaki
```

The server prompts you to enter a password for the user account you specify with the --user option. (If you omit the --user option, you will first be prompted to enter a user ID.)

The server is started when a message similar to the following appears:

```
2003-07-31 19:57:18,458 INFO
[org.jboss.system.server.Server, main] JBoss (MX
MicroKernel) [3.0.6 (CVSTag=JBoss_3_0_6 Date=200301260037)]
Started in 0m:27s:139ms
```

**In Unix:** The share server startup script is in the Avaki installation directory (such as /root/AvakiDataGrid70). You can start a share server with or without the auto-restart feature; both methods are described here.

- To start the share server with auto-restart enabled, log in as root, switch to your Avaki installation directory and issue a command of this form:

```
share-server --register [--user=<name>]
```

- To start the share server without enabling auto-restart, switch to your Avaki installation directory and issue the following command:

```
$ share-server --start
```

The server is started when a message similar to the following appears:

```
2003-07-31 19:57:18,458 INFO
[org.jboss.system.server.Server, main] JBoss (MX
MicroKernel) [3.0.6 (CVSTag=JBoss_3_0_6 Date=200301260037)]
Started in 0m:27s:139ms
```

- Step 6** Install an SSL certificate on the share server. For instructions, see [“Installing an SSL certificate” on page 39](#).

- Step 7** In a browser, log in to the grid domain as a member of the Administrators group and navigate to the Connect Share Server screen:

Home > Server management > Connect new share server

**Connect Share Server**

To connect a share server to a grid server, install the share server on the machine where the server will reside, start the share server, and then specify the server's logical name, IP or DNS name, connect port, and the grid server to which the share server will be joined.

Logical name for new share server:

IP or DNS name of new share server:

Connect port of new share server:

Grid server to join to:

- Step 8** Fill in the form:

- Logical name for new share server: Enter a name for the share server.
- IP or DNS name of new share server: Enter the IP address or DNS name of the machine the share server is running on.
- Connect port of new share server: Enter 2099.
- Grid server to join to: From the pull-down menu, select the name of the grid server (or GDC) with which this share server will be associated.

---

**Note** If you need to set up a share server using ports other than the defaults, including 2099, follow the steps in the procedure that follows, “[Configuring multiple share servers on one machine OR configuring share servers to use nondefault ports.](#)” (Execute the procedure just once to create a single share server using the ports you specify.)

---

**Note** You can find an Avaki server’s connect port (as well as its name) in its connectinfo.txt file, which is generated when the server starts. The path to this file varies by server type:

<Avaki-install-dir>/jboss/server/share-server/log/connectinfo.txt

(<Avaki-install-dir> is the local directory where Avaki software is installed.)

**Step 9** Click **Submit**.

## Configuring multiple share servers on one machine OR configuring share servers to use nondefault ports

To set up several stand-alone share servers to run on the same machine, or to set up one or more share servers that use ports other than the default ports, do the following for each share server.

---

**Note** Do not reassign ports on an Avaki server once it’s been joined to an Avaki domain; doing so will cause communication problems between servers. If the server has been joined to a domain, you must re-install and reconfigure the server, change the ports, and join the new server to the Avaki domain.

**Step 1** Run the Avaki installer on the target machine and select a Share Server installation. For more instructions, see “[Installing in Unix](#)” on page 16 or “[Installing in Windows](#)” on page 22. Installation notes:

- Don’t install the second share server into the same directory you used for the first installation on this machine. For example, you might call the two installation directories AvakiShareServer1 and AvakiShareServer2.
- If you’re installing on a Windows machine, don’t create shortcuts for the second share server in the same program group you used for the first. We suggest that you name the two program groups after their respective install directories—for example, AvakiShareServer1 and AvakiShareServer2.

- Step 2** Edit the `bindings.xml` file in the `/jboss/server/share-server/conf` subdirectory of the Avaki installation directory created in [Step 1](#). Six ports are specified in this file: 2098, 2099, 9083, 6444, 9443, and 9080. Replace as many port numbers as necessary with values appropriate for this machine. Carefully record the port number you enter in place of 2099 (the second port number in the file); this is the share server's connect port, and you must supply the connect port number when you connect this share server to the grid domain in [Step 9](#), below.

---

**Note** Two RMI ports are linked to the connect port, which defaults to 2099. These port numbers are determined by adding 1 and 2 to the connect port number—for example, when you use the default connect port number, 2099, the linked RMI ports are 2100 and 2101. When you select port numbers, be sure that the two ports after the connect port are also unused. (It is not possible to set the linked RMI port numbers independently of the connect port number.)

- Step 3** Edit the `shareserver.ports` file in the `/jboss/server/share-server/conf` subdirectory of your Avaki installation directory. Two ports are specified in this file: 9510 and 9511. If necessary, replace these port numbers with values appropriate for this machine. (We suggest using two consecutive available ports.)
- Step 4** (Optional) Specify a host name or IP address for your share server to advertise if the default (the server machine's name) is unsuitable. For instructions, see [“Setting a server's host name or IP address”](#) on page 32.
- Step 5** If you have chosen to generate self-signed SSL certificates for your Avaki servers, set up this share server to do so. For instructions, see [“Generating self-signed SSL certificates”](#) on page 34. Note that you must perform this step on each share server separately.
- Step 6** Start the share server. For instructions, see [Step 5](#) in the previous procedure—but note that if you're using Windows shortcuts, you need to look for them in the program group you created when you installed, as noted in [Step 1](#) above.
- Step 7** Install an SSL certificate on the share server. For instructions, see [“Installing an SSL certificate”](#) on page 39.



**Step 8** In a browser, navigate to the Connect Share Server screen:

Home > Server management > Connect new share server

**Connect Share Server**

To connect a share server to a grid server, install the share server on the machine where the server will reside, start the share server, and then specify the server's logical name, IP or DNS name, connect port, and the grid server to which the share server will be joined.

Logical name for new share server:

IP or DNS name of new share server:

Connect port of new share server:

Grid server to join to:  ▾

**Step 9** Fill in the form:

- Logical name for new share server: Enter a name for the share server. (You must specify this name when you create an Avaki share using this share server.)
- IP or DNS name of new share server: Enter the IP address or DNS name of the machine the share server is running on.
- Connect port of new share server: Enter the port number you entered in place of 2099 in [Step 2](#).
- Grid server to join to: From the pull-down menu, select the name of the grid server (or GDC) with which this share server will be associated.

**Step 10** Click **Submit**. The system connects the new share server to the Avaki domain.

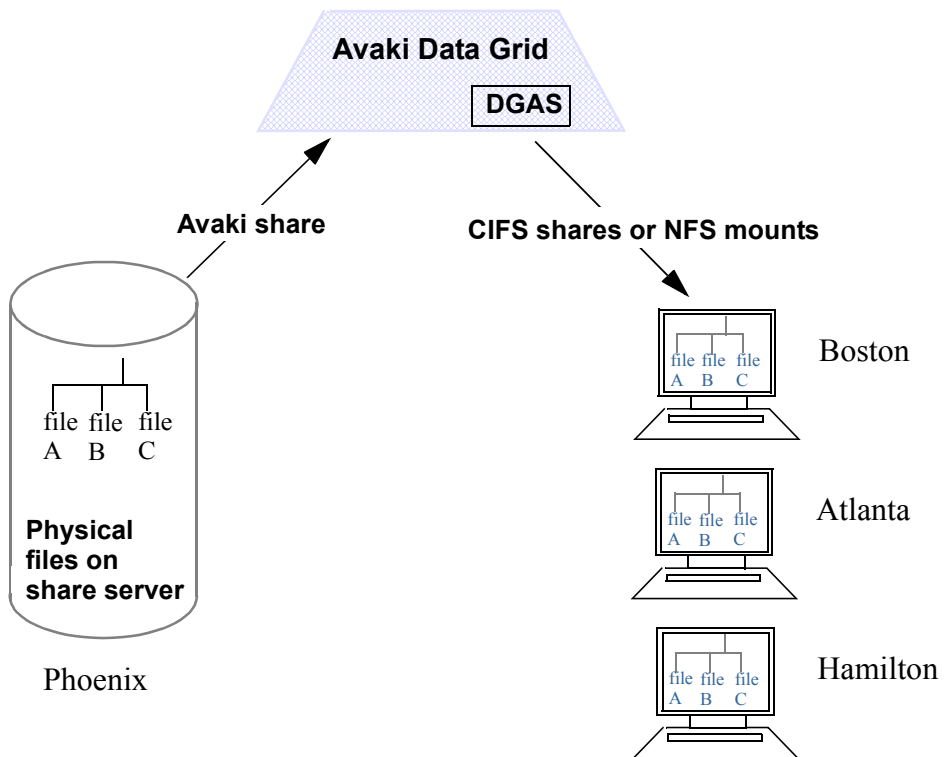
For instructions on setting up Avaki shares, see [“Creating Avaki shares” on page 258](#).

# Setting up data grid access servers

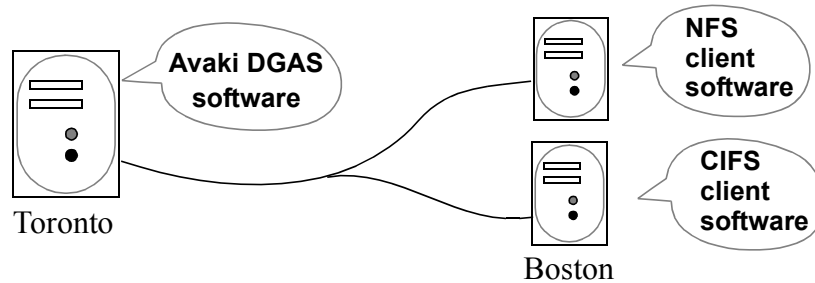
## About data grid access servers

Client computers that are not running Avaki software can access Avaki directories through a data grid access server (DGAS). Users of the client computers can read, write and delete mounted files as if they were local. Users also benefit from local caching of mounted files.

This illustration shows files in Phoenix being shared into the data grid (which includes a DGAS) and viewed in Boston, Atlanta, and Hamilton.



This illustration shows a data grid computer in Toronto connecting to client machines in Boston. The computer in Toronto is running a DGAS and the clients are running NFS client software, which is typically included with Unix, or CIFS client software, which is built into Windows. (There is no Avaki software on the clients.)



When you're configuring a DGAS and its NFS and CIFS clients, it's important to keep several distinctions clear:

- DGAS machine and client machine(s).
- NFS servers and NFS clients
- CIFS servers and CIFS clients
- Data grid user accounts and local (or directory service) user accounts

Consider making a drawing of the DGAS configuration you are planning. Refer to the drawing as you follow the DGAS set-up procedures.

The procedure for setting up data grid access servers begins here. The client set-up procedures are:

- Setting up NFS clients to mount data grid directories, [page 92](#).
- Setting up CIFS clients to access data grid directories, [page 94](#).

## System requirements for data grid access servers

The machine that will be the data grid access server (which need not be the grid domain controller) must:

- Meet the DGAS requirements listed in the section [“Loading and starting Avaki Data Grid software” on page 15](#) (including operating system, minimum memory, and minimum disk space), and
- Have Avaki software installed. Perform a Data Grid Access Server installation (see [“Loading and starting Avaki Data Grid software” on page 15](#)).

---

**Note** We recommend against running a DGAS on a Unix machine where file locking in is use. The DGAS replaces the entry for the NLM protocol in the RPC port mapper, which prevents the native lockd from functioning.

---

**Note** If you plan to write large files (1GB or greater) through this DGAS from an NFS client, we recommend against running the DGAS on the same machine as the NFS client. Writing large files through a DGAS running on the same machine as an NFS client can expose OS-level deadlocks.

## Overview of the data grid access server setup procedure

You must complete the following tasks:

1. Disable any NFS or CIFS server software on the machine where the data grid access server will run ([“Preparing to run a data grid access server” on page 65](#)).
2. Create or import any user accounts or user groups required in the grid domain and in the external authentication service that will authenticate users accessing the grid domain from the CIFS or NFS clients ([“Setting up user accounts and groups” on page 68](#)).
3. Map UIDs and GIDs from external authentication services to user accounts and groups that have been imported into (or created in) the grid domain ([“Setting domain-wide user and group mappings” on page 68](#)).
4. (Optional) Specify a host name or IP address for your grid server to advertise if the default (the server machine’s name) is unsuitable. For instructions, see [“Setting a server’s host name or IP address” on page 32](#).
5. Use the `dgas` command to start the data grid access server ([“Starting the data grid access server” on page 73](#)).

6. Use the web UI to connect the new DGAS to an Avaki domain (“[Connecting the DGAS to your Avaki domain](#)” on page 79).
7. (Optional) Use the web UI to configure DGAS properties (“[Configuring DGAS properties](#)” on page 82). When you configure properties, you may find it helpful to consult someone who’s familiar with the requirements of the applications that will access data through the DGAS.
8. (Optional) If you configured any nondynamic properties, restart the DGAS (“[Restarting the DGAS](#)” on page 85).
9. Configure admission policies for hosts and networks (“[Setting admission policies](#)” on page 86).
10. (Optional) Configure per-DGAS user and group mappings (“[Setting per-DGAS user and group mappings](#)” on page 88). For details on how mappings work, see [Appendix B, “DGAS access effects”](#).
11. (Optional) Configure the DGAS to use a cache service on an Avaki grid server (“[Configuring a cache service for a DGAS](#)” on page 90).
12. Mount selected grid directories from one or more NFS or CIFS client machines (“[Setting up NFS clients](#)” on page 92 and “[Setting up CIFS clients](#)” on page 94).

The sections that follow provide instructions for these tasks.

## Preparing to run a data grid access server

On a machine that will be configured with a data grid access server, the DGAS takes the place of any native NFS or CIFS server. This section explains how to disable the native NFS server (“[On a machine that will serve NFS clients,](#)” below) or CIFS server (“[On a machine that will serve CIFS clients](#)” on page 67).

### On a machine that will serve NFS clients

Most operating systems allow only one NFS server daemon to hold a port mapping at a time. Avaki data grid access servers essentially replace native NFS servers; they speak the NFS client protocol and therefore can’t, without adjustment, coexist with NFS daemons on operating systems where only one NFS daemon is allowed.

If an NFS daemon is running on your machine, you must shut down the NFS daemon and remove its port mapping. If you’re working on a Windows machine, refer to your NFS documentation. If you’re working on a Unix machine, see the procedure below for instructions.

---

**Note** This section directs you to shut down NFS server daemons running on the machine that will host your DGAS. (Later, you will be directed to install NFS client software on machines that will mount Avaki directories through the DGAS.)

**Finding and shutting down native NFS daemons in Unix.** To determine whether a native NFS daemon is present, enter `rpcinfo -p`, followed by the machine name. If the resulting display shows no nfs entries, skip to the section [“Setting up user accounts and groups”](#) on page 68.

If the resulting display includes nfs and mountd entries similar to the following, this machine is running an NFSd.

```
100003 3 udp 4788 nfs
100005 3 udp 4788 mountd
```

If you choose to shut down any native NFS daemons and remove their port mappings before starting a data grid access server, follow these steps.

**Step 1** Log in to the machine as root.

**Step 2** Enter `rpcinfo -p <machine-name>`. In the resulting output, note the numbers in the first two columns of the lines for the NFS daemons and mountds. The first column gives the program number; the second gives the NFS version number.

```
$ rpcinfo -p Flintstones
100003 3 udp 4788 nfs
100005 3 udp 4788 mountd
100003 2 tcp 3206 nfs
100005 1 tcp 3206 mountd
```

**Step 3** Use the `ps` command to find the process IDs for all the NFS and mountd processes running on the machine.

**Step 4** Use the `kill -2` command to shut down all the NFS and mountd processes.

**Step 5** To remove the port mappings, enter `rpcinfo -d <program#> <version#>` commands for each NFS daemon and mountd mapping. Use the program and version numbers you discovered in [Step 2](#).

```
$ rpcinfo -d 100003 3
$ rpcinfo -d 100005 3
$ rpcinfo -d 100003 2
$ rpcinfo -d 100005 1
```

## On a machine that will serve CIFS clients

If the DGAS will run on a Windows machine and if you want the DGAS to serve CIFS shares, disable native CIFS Windows services running on ports 445, 137, 138, and 139.

**Caution** If you disable CIFS on these ports, the machine will be unable to use network printers, access network drives on other machines, or share files out to other machines.

Follow these steps:

- Step 1** To disable ports 137, 138 and 139, first open the Windows Control Panel and select Network and Dialup Connections.
- Step 2** Open the connection that the machine uses to connect to the data grid, for example, Local Area Connection. (Unless the machine has more than one network card, there will be only one choice.)
- Step 3** Click the **Properties** button.
- Step 4** Open Internet Protocol (TCP/IP).
- Step 5** Click **Advanced**.
- Step 6** Select the WINS tab.
- Step 7** Select Disable NetBIOS over TCP/IP.
- Step 8** Click **OK**. Ports 137, 138 and 139 are released by CIFS Windows services.
- Step 9** To release port 445, use the regedit tool to add the following value to the registry:  
 Key: HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\NetBT\Parameters  
 Value: SmbDeviceEnabled  
 Type: DWORD value (REG\_DWORD)  
 Content: 0 (to disable)
- Step 10** Reboot the machine so the registry edit will take effect.

## Setting up user accounts and groups

Make sure that all the users who will access the data grid through this DGAS have user accounts that have been created in or imported into the grid, or belong to groups have been created in or imported into the grid.

In the section that follows, “[Setting domain-wide user and group mappings](#),” you will create mappings between the grid accounts and groups for these users and the login accounts that they use on their NFS client machines. When these steps are complete, users will be able to log in to their NFS clients and read and write grid data as if they were logged in to the grid domain.

If you plan to configure default mappings, which allow unmapped users to access the data grid, set up a special default mapping user account for this purpose. You may also choose to set up a special default mapping group. You can create the default mapping user and group in the grid, or create them in an NIS or LDAP service that has been integrated with Avaki and import them into the grid.

These procedures may be useful:

- “[Importing user accounts from an LDAP authentication service](#)” on page 157
- “[Importing groups from an LDAP authentication service](#)” on page 159
- “[Importing user accounts from an NIS authentication service](#)” on page 163
- “[Importing groups from an NIS authentication service](#)” on page 165
- “[Creating grid user accounts](#)” on page 168
- “[Creating grid groups](#)” on page 191

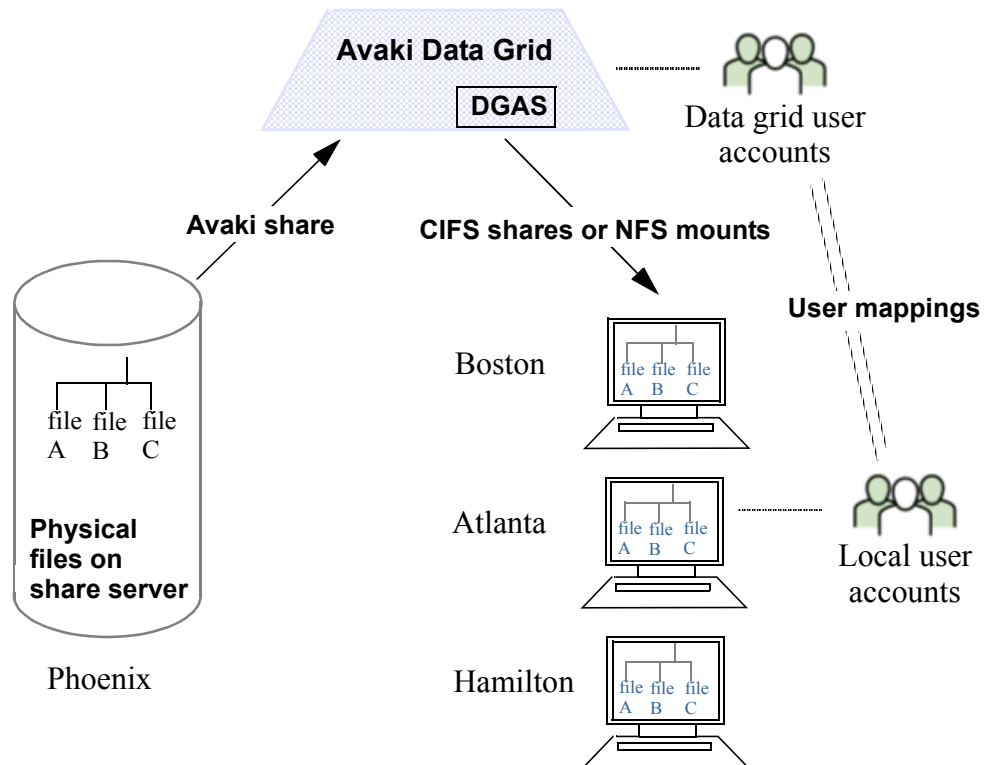
## Setting domain-wide user and group mappings

Two sets of user and group mappings affect DGAS behavior:

- Domain-wide mappings, which are discussed here.
- DGAS-specific mappings, which override domain-wide mappings. For information on DGAS-specific mappings, see “[Setting per-DGAS user and group mappings](#)” on page 88.



A data grid access server allows you to view and modify files and directories across file system boundaries. In the illustration below, for example, users in Atlanta can edit files in a remote file system in Phoenix via the data grid. Because file systems permit changes only by known users, it is necessary to set up correspondences, or mappings, between grid user accounts and local (authentication service) user accounts, and between grid groups and local groups. (In the mappings, local Unix user accounts and groups are identified by their numeric user IDs (UIDs) and group IDs (GIDs).)



For example, suppose you have configured a mapping between local user Fred in Atlanta and data grid user Fred. Local user Fred wants to edit a file in Phoenix. Local user Fred does not have permission to edit the file, but data grid user Fred does. Because of the mapping, local user Fred's changes are attributed to data grid user Fred and accepted.

**Default mappings.** You may choose to let unmapped users access the data grid with restricted privileges by associating unmapped users with a special default grid account or group. This is called a default mapping. Any authentication service can be configured with a default user or group so that any unmapped user can gain access to the grid as the default user or as members of the default group. If no default user or group is mapped, unmapped users cannot access data grid resources through NFS or CIFS.

The procedures in this section (below) explain how to set up user account mappings, group mappings, and default mappings. For further details on how mappings work, see [Appendix B, “DGAS access effects”](#).

## Setting up user account mappings

Follow these steps.

- Step 1** Get a list of UIDs for the NFS client users who will access the data grid through this data grid access server.
- Step 2** Log in as a member of the UserAdministrators or Administrators group.
- Step 3** Navigate to the Select User screen:  
Home > User management > View and modify users
- Step 4** Click boxes in the Select column to select the users for whom you want to create mappings.
- Step 5** Click **Update UIDs**. The Update UIDs screen appears.

**Update UIDs**

To configure the user ID (UID) that is mapped to a user account, enter the UID in the New UID box for the user.

| User Name | Authentication Service | Authentication Service Type | New UID                                   |
|-----------|------------------------|-----------------------------|-------------------------------------------|
| fred      | DefaultAuthService     | Grid                        | <input style="width: 100%;" type="text"/> |
| wilma     | DefaultAuthService     | Grid                        | <input style="width: 100%;" type="text"/> |

- Step 6** In the New UID box for each user account, enter the UID that you are mapping to the account.

---

**Note** Map each UID to only one user.

- Step 7** Click **Update UIDs** to save the mappings. The system displays a confirmation page showing that the links have been created.

## Setting up group mappings

Follow these steps.

- Step 1** Get a list of GIDs for the NFS client groups whose users will access the data grid through the data grid access server.
- Step 2** Log in as a member of the UserAdministrators group.
- Step 3** Navigate to the Select Group screen:  
Home > User management > View and modify groups
- Step 4** Click boxes in the Select column to select the groups for which you want to create mappings.
- Step 5** Click **Update GIDs**. The Update GIDs screen appears.

**Update GIDs**

To configure the group ID (GID) that is mapped to a group account, enter the GID in the New GID box for the group.

| Group Name  | Authentication Service | Authentication Service Type | New GID                                   |
|-------------|------------------------|-----------------------------|-------------------------------------------|
| Bedrock     | DefaultAuthService     | Grid                        | <input style="width: 100%;" type="text"/> |
| DomainUsers | DefaultAuthService     | Grid                        | <input style="width: 100%;" type="text"/> |

- Step 6** In the New GID box for each group, enter the GID that you are mapping to the group.

---

**Note** Map each GID to only one group.

- Step 7** Click **Update GIDs** to save the mappings. The system displays a confirmation page showing that the links have been created.

## Setting up default mappings

Follow these steps.

- Step 1** Log in to Avaki as a member of the Administrators group.
- Step 2** Navigate to the View Authentication Services screen:  
Home > User management > Manage existing authentication services

- Step 3** Click the **View/Edit** link next to the desired authentication service. The NFS Mapping Information screen appears.

### NFS Mapping Information

If a data grid access server has been configured for your grid domain, users and groups in the grid domain's authentication service must be mapped to users and groups on the NFS client(s) that mount grid directories. If you have not configured specific mappings in your authentication service for every user and group that might access grid data through the data grid access server, configure default mappings below. The data grid access server uses the default mappings when no specific mapping exists for a user or a group.

|                            | Current     | New                                                                  |
|----------------------------|-------------|----------------------------------------------------------------------|
| <b>Default UID:</b>        | unspecified | <input style="width: 100%;" type="text" value="0"/>                  |
| <b>Default GID:</b>        | unspecified | <input style="width: 100%;" type="text" value="0"/>                  |
| <b>Default user name:</b>  | unspecified | <input style="width: 100%;" type="text" value="defaultUserAccess"/>  |
| <b>Default group name:</b> | unspecified | <input style="width: 100%;" type="text" value="defaultGroupAccess"/> |

- Step 4** In the New column, enter the following (or leave any field blank to leave the item unspecified):
- **Default UID:** The DGAS uses this UID to display the ownership and permissions of files owned by unmapped users.
  - **Default GID:** The DGAS uses this GID to display the group ownership and permissions of files owned by unmapped groups.
  - **Default user name:** The DGAS uses the credentials of the named user in this authentication service to access files when no explicit mapping is found in this DGAS or authentication service.
  - **Default group name:** The DGAS uses the credentials of the named group in this authentication service to access files when no explicit mapping is found in this DGAS or authentication service.

---

**Note** In contrast to default user and group names, which are used to permit or deny access, the default UID and GID are used for display only, and on most operating systems do not affect user access.

- Step 5** Click **Submit** to save the default mapping you have created.

## Changing the host name

(Optional) Specify a host name or IP address for your DGAS to advertise if the default (the DGAS machine's name) is unsuitable. For instructions, see [“Setting a server's host name or IP address” on page 32](#).

## Starting the data grid access server

In the procedure that starts on [page 74](#), you use the **dgas** command to initialize a data grid access server. Here is the command syntax, followed by descriptions of the options:

```
dgas --start [--name=<server-name>] [--db-path=<local-path>]
 [--cache-path=<local-path>] [--new] [--port=<connect-port>]
 [--rmi-port=<rmi-server-port>]
```

```
dgas --register [--user=<user>] [--name=<server-name>]
 [--db-path=<local-path>] [--cache-path=<local-path>]
 [--port=<connect-port>] [--rmi-port=<rmi-server-port>]
```

You can also use **dgas** to shut down or unregister a data grid access server:

```
dgas --stop [--name=<dgas-name>]
```

```
dgas --unregister [--name=<server-name>]
```

|                                              |                                                                                                                                                          |
|----------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>--start</code>                         | Initialize a DGAS (no auto-restart).                                                                                                                     |
| <code>--register</code>                      | Initialize a DGAS and register it as a Unix or Windows service. When you use this option, the DGAS restarts automatically whenever its machine restarts. |
| <code>--name=&lt;server-name&gt;</code>      | (Optional) Specify a name for the data grid access server. If you don't specify a name for a new DGAS, the system uses the default name Access1.         |
| <code>--user=&lt;user&gt;</code>             | Specify the user account under which the data grid access server will run.                                                                               |
| <code>--cache-path=&lt;local-path&gt;</code> | (Optional) Specify the location of the local directory that holds DGAS internal caches.<br>Default: <Avaki-install-directory>/DGAS/cache                 |

- `--db-path=`  
`<local-path>` (Optional) Specify the location of the DGAS state database, which stores property settings and other information required to restart the DGAS.  
Default: `<Avaki-install-directory>/DGAS/dgas_db`
- `--new` (Optional) Wipe out the state database for the previous DGAS but retain any DGAS internal caches. Don't use this option when you use **dgas --register**.
- `--port=`  
`<connect-port>` (Optional) Specify the connect port that the DGAS will use to communicate with other objects in the data grid.  
Default value: 1399
- `--rmi-port=`  
`<rmi-server-port>` (Optional) Specify the RMI server port that the DGAS will use.  
Default value: 1599

Before starting this procedure, install Avaki software on the machine where the data grid access server will run, as described in [“Loading and starting Avaki Data Grid software” on page 15](#). Be sure to select the Data Grid Access Server installation option. Follow these steps to start a DGAS:

- Step 1** Log in to the machine as follows:
- Unix login requirements: Log in as root if any of the following conditions apply:
    - this DGAS will be registered as a service (for auto-restart of the DGAS when the machine reboots)
    - this DGAS will serve CIFS shares
    - this DGAS will use TCP or UDP ports numbered lower than 1024
  - Windows login requirements: If you will register this DGAS as a service so that it restarts automatically when the machine reboots, log in as a user who has permission to log in as a service.
- Step 2** Change to the Avaki installation directory. (That is, the local directory where Avaki software was installed.)
- Step 3** Log in to your grid domain as a member of the Administrators group.
- Step 4** Use the **dgas** command to start the data grid access server.

In the following example,

- The `--register` option specifies that this DGAS will be registered as a service with the operating system and will restart automatically whenever the machine it runs on reboots.
- The `--user` option specifies the user account under which the DGAS runs. (You will be prompted for the password.)
- The `--name` option assigns the name “`dgas12`” to this DGAS.
- The `--cache-path` option specifies the location of the directory that holds all DGAS caches.
- The `--db-path` option specifies the location of the DGAS state database, which stores property settings and other information required to restart the DGAS.

```
c:\ dgas --register --user=avaki --name=dgas12
--cache-path="c:\AvakiDataGrid70\cache"
--db-path="c:\AvakiDataGrid70\dgas_db"
```

---

**Note** If you want the DGAS to use ports other than the defaults, use the `--port` option to specify the connect port number and the `--rmi-port` option to specify the RMI server port number. (The defaults are 1399 and 1599, respectively.)

**Step 5** To ensure that the data grid access server is running, enter `rpcinfo -p`. The resulting display should include `nfs` and `mountd` entries similar to the following:

```
100003 3 udp 4788 nfs
100005 3 udp 4788 mountd
100003 3 tcp 1856 nfs
100005 3 tcp 1856 mountd
100003 2 udp 4788 nfs
100005 1 udp 4788 mountd
100005 2 udp 4788 mountd
100003 2 tcp 1856 nfs
100005 1 tcp 1856 mountd
100005 2 tcp 1856 mountd
```

See “[Configuring the DGAS to use nondefault ports](#),” below, if you need to specify particular ports for the DGAS to use. Otherwise, proceed directly to “[Connecting the DGAS to your Avaki domain](#)” on page 79.

## Configuring the DGAS to use nondefault ports

If the ports that DGASes use by default conflict with ports already in use on the DGAS machine, you can specify other ports for your DGAS to use. (For a list of default ports, see [“Planning use of network ports” on page 8.](#))

---

**Note** Do not reassign ports on an Avaki server once it’s been joined to an Avaki domain; doing so will cause communication problems between servers. If the server has been joined to a domain, you must re-install and reconfigure the server, change the ports, and join the new server to the Avaki domain.

There are two ways to configure a DGAS to use nondefault ports:

- In the method described in this section, you create and load a DGAS properties file that contains the desired port settings. This method works well when you’re starting up the DGAS for the first time.
- The second method, which is easiest when your DGAS is already running, is to change property values in the CLI with the **avaki dgas --set-property** command, or in the web UI on the Configure DGAS screen (Home > Server management > View data grid access servers > Configure). Changes made this way do not take effect until you restart the DGAS.

---

**Note** This section does not cover two default ports: the connect port, which defaults to 1399, and the RMI server port, which defaults to 1599. To change these ports, use **dgas** command options at start-up, as described in [“Starting the data grid access server” on page 73.](#)

### Creating a DGAS properties file

To change the default ports, you can create a properties file for the DGAS. The properties file sets values for the DGAS properties you want to change, and the DGAS uses the default values for properties that don’t appear in the properties file. The table below lists DGAS properties that affect port usage; following the table is a sample DGAS properties file that sets the properties in the table.



| Property            | Description                                                                                                                                                                                                                                                                                                                                                       |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| cifs-enabled        | Specifies whether this DGAS can serve CIFS shares. If you enable CIFS, you must also enable SMB over NetBIOS (netbios-smb-enabled), TCP/IP (tcpip-smb-enabled), or both.<br><br>Values: true or false<br>Default: true                                                                                                                                            |
| mount-port          | Specifies the port on which the NFS mount server listens.<br><br>Values: valid port numbers<br>Default: 0 (the DGAS chooses an available port)                                                                                                                                                                                                                    |
| netbios-smb-enabled | Enables or disables the SMB protocol over NetBIOS (ports 137 to 139), which supports CIFS shares.<br><br>Values: true or false<br>Default: true                                                                                                                                                                                                                   |
| nfs-port            | Specifies the port on which NFS clients will connect to the DGAS over TCP or UDP. If you specify an NFS port other than 2049, the default, you must include the port number in the <b>mount</b> command you issue from the client. (Note, however, that some clients don't allow you to specify a mount port.)<br><br>Values: valid port numbers<br>Default: 2049 |
| nlm-port            | Specifies the port on which the NLM server listens.<br><br>Values: valid port numbers<br>Default: 0 (the DGAS chooses an available port)                                                                                                                                                                                                                          |
| tcpip-smb-enabled   | Enables or disables the SMB protocol over TCP/IP (port 445), which supports CIFS shares.<br><br>Values: true or false<br>Default: true                                                                                                                                                                                                                            |

This sample DGAS properties file sets the mount, NLM, and NFS ports to nondefault values. It also disables CIFS/SMB, freeing ports 137, 138, 139 and 445. (It is not possible to assign the CIFS/SMB functions to other ports—if you need those ports for other purposes, you must disable CIFS/SMB.)

```
mount-port = 11500
nlm-port = 11501
nfs-port = 11502
```

```
netbios-smb-enabled = false
tcpip-smb-enabled = false
cifs-enabled = false
```

Save your DGAS properties file in a local directory on the machine where the DGAS runs. (The properties file is not actually required to be on the DGAS machine; it must be on the machine where you execute the **avaki dgas --initialize** command that loads the file, as described in the section that follows.)

## Using the DGAS properties file

This procedure assumes that your DGAS has been started, but has not been connected to an Avaki domain.

Follow these steps to use the DGAS properties file to change the default ports:

**Step 1** Create a properties file as described in [“Creating a DGAS properties file” on page 76](#).

**Step 2** On the DGAS machine, do a client connect to GDC:

```
avaki client --connect <GDC-machine>
```

where <GDC-machine> is the name or IP address of the machine that serves as the grid domain controller for this Avaki domain. For example:

```
$ avaki client --connect Flintstone
```

**Step 3** Log in to Avaki as a member of the Administrators group and change to the Avaki installation directory.

**Step 4** To connect the DGAS to the Avaki domain, enter a command of this form:

```
avaki server --connect <DGAS-machine>[:<port>] <DGAS-name>
```

where <DGAS-machine> is the DNS name or IP address of the machine where the DGAS is running,

<port> is the connect port of the DGAS (required only if you assigned a nondefault connect port), and

<DGAS-name> is the name you are assigning to this DGAS.

For example:

```
avaki server --connect Slate:1501 Dgas1
```

**Step 5** To make the DGAS read the properties file, enter a command of this form:

```
avaki dgas --initialize <DGAS-name>
 --properties-file=<local-path>
```

where <DGAS-name> is the name you assigned to this DGAS in [Step 4](#), and

<local-path> is the path to your DGAS properties file.

For example:

```
avaki dgas --initialize Dgas1
 --properties-file=/home/fred/dgas1.properties
```

This procedure connects the DGAS to your existing grid domain. After completing it, skip to [“Configuring DGAS properties” on page 82](#).

## Connecting the DGAS to your Avaki domain

Follow these steps to connect a DGAS that you have initialized with the **dgas** command:

**Step 1** Open a web browser and point it to

```
http://<gdc-name>:7080
```

where <gdc-name> is the name of the grid domain controller for the Avaki domain to which you’re joining the new DGAS.

**Step 2** Log in to Avaki as a member of the Administrators group.

**Step 3** Navigate to the Connect DGAS screen:

Home &gt; Server management &gt; Connect new data grid access servers

### Connect DGAS

To connect a DGAS to the grid, install the DGAS on the machine where it will reside, start it, provide the information below, and click **Submit**. Note: for most use cases, both NetBIOS SMB and TCP/IP SMB should be enabled when CIFS is enabled.

Logical name for new DGAS:

IP or DNS name of new DGAS:

Connect port of new DGAS:

Cache block size (in bytes):

Cache fragments per block:

NFS port:

Mount port:

Update port mapper?

Lock manager enabled?

Mount server enabled?

NFS V3 enabled?

NFS V2 enabled?

CIFS enabled?

NetBIOS SMB enabled?

TCP/IP SMB enabled?

---

**Note** Review the options on this screen with someone who's familiar with the requirements of the applications that will access data through this DGAS.

**Step 4** Fill in the form:

**Logical name for new DGAS:** Enter a name to identify the DGAS. If you used the **dgas** command's **--name** option to assign a name to the DGAS, the name you enter here must be the same. If you did not use the **--name** option, enter the default name, `Server1`.

**IP or DNS name of new DGAS:** Enter the IP address or fully qualified DNS name of the machine on which the new DGAS is running. (For example: `porcupine.sybase.com`.)

**Connect port of new DGAS:** if you used the **dgas** command's **--port** option to specify a connect port other than 1399, the default, enter that other port number here. Otherwise, leave the default number in place.

**Cache block size:** Specify the block size of the DGAS per-file read cache in bytes. The cache for each file is broken into equal-sized blocks, which in turn are broken into equal-sized fragments. The cache fragments per block field, below, specifies the number of fragments per block.

A large block size improves performance by reducing the number of times the DGAS must open and close files.

Note that the cache block size and cache fragments per block values act together to affect cache behavior. If you change one value, consider how the change will affect the other.

Recommended range: 10240 to 209715200 bytes (10 MB to 200 MB)

**Cache fragments per block:** Specify the number of fragments each block of the DGAS read cache is divided into. The cache block size value must be divisible by the cache fragments per block value so that each fragment is equal in size. The fragment size value (cache block size / cache fragments per block) is also used for two other purposes: the size of read-ahead data that the DGAS reads from the back end files as needed, and the size of in-memory cache for each cached file. The recommended range for fragment size is 256K to 2MB; calculate cache block size and cache fragments per block values with this range in mind. The main constraint on fragment size is memory. If you have lots of memory, you can afford a larger fragment size.

Note that the cache block size and cache fragments per block values act together to affect cache behavior. If you change one value, consider how the change will affect the other.

Recommended range: 100 to 800 fragments

**NFS port:** The port on which NFS clients will connect to the DGAS over TCP or UDP. If you specify an NFS port other than 2049, the default, you must include the nondefault port number in the **mount** command you issue from the client. (Note, however, that some clients don't allow you to specify a mount port.)

**Mount port:** The port on which the NFS mount server listens. If this option is set to 0, the default, the DGAS chooses an available port. This option has no effect on CIFS clients.

**Update port mapper:** Indicate whether the DGAS registers its RPC services with the port mapper. Turn this off (by removing the check mark) only if you intend to run a native NFS daemon on the same machine as the DGAS.

**Lock manager enabled?:** Specify whether advisory file locking is enabled. The lock manager protocol provides file locking support for files accessed through the data grid access server. Enable this option (check the box) if an application that will access data through this DGAS requires file locking. If you plan to connect this DGAS to NFS clients that use Windows Services for Unix (SFU), you should either enable this option or use the `-o nolock` mount option. This option has no effect on CIFS clients.

**Mount server enabled?:** For NFS, specify whether the mount protocol is enabled. This option has no effect on CIFS clients.

**NFS V3 enabled?:** Specify whether the NFS version 3 protocol is enabled.

**NFS V2 enabled?:** Specify whether the NFS version 2 protocol is enabled. (You can enable both V2 and V3.)

**CIFS enabled?:** Specify whether CIFS is enabled. Enable CIFS only if you plan to serve CIFS shares from this DGAS.

**NetBIOS SMB enabled?:** Specify whether to enable the SMB protocol over NetBIOS (ports 137 to 139), which supports CIFS shares. Enable NetBIOS SMB only if you plan to serve CIFS shares from this DGAS.

**TCP/IP SMB enabled?:** Specify whether to enable the SMB protocol directly over TCP/IP (port 445), to support CIFS shares. Enable TCP/IP SMB only if you plan to serve CIFS shares from this DGAS.

- Step 5** Click **Submit** to connect the new DGAS to the Avaki domain. The system displays the View DGASes screen, listing the new DGAS in the Server Name column.

From this screen, you can access the configuration screens used in the procedures that follow.

## Configuring DGAS properties

You can set a large number of properties that control how the DGAS behaves. These properties affect the DGAS caches, asynchronous writes, NFS, CIFS, and other options. The properties are described in the Configure DGAS screen in the web UI.

It is not necessary to change property settings for every DGAS; a DGAS often works well using the default settings. However, you might find it helpful to review the prop-

erties and understand how they work, even if you don't reset any of them. We recommend that you review the properties with someone who thoroughly understands the requirements of the applications that will access data through this DGAS.

Some DGAS properties are dynamic. (Dynamic properties are identified in the “Dynamic?” column of the Configure DGAS screen shown in [Step 3](#), below.) You can change the value of a dynamic property while the DGAS is running. If you change the value of a nondynamic property, you must restart the DGAS to make the change take effect.

As you perform the procedure below, make a note if you change the value of any dynamic property. If you do, be sure to restart the DGAS, as described the next procedure, “[Restarting the DGAS](#)” on page 85.

Follow these steps to set DGAS properties.

- Step 1** Log in to Avaki as a member of the Administrators group.
- Step 2** Navigate to the View DGASes screen:

Home > Server management > View DGASes

**View DGASes**

You are viewing the DGASes in the local grid domain.

| Server Name | Configure                 | Cache                 | Mappings                 | Policy                 | Destroy                 | Attributes                 | Security                 |
|-------------|---------------------------|-----------------------|--------------------------|------------------------|-------------------------|----------------------------|--------------------------|
| dgas1       | <a href="#">Configure</a> | <a href="#">Cache</a> | <a href="#">Mappings</a> | <a href="#">Policy</a> | <a href="#">Destroy</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

- Step 3** Click the **Configure** link for the DGAS you want to configure. The system displays the Configure DGAS screen, of which this is part:

| Configure DGAS               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |          |                                   |
|------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|-----------------------------------|
| DGAS Properties: dgas1       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |          |                                   |
| Property Name                | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Dynamic? | Value                             |
| async-write-blocks-pool-size | This property specifies the number of blocks in the pool of memory blocks used for asynchronous writes. The DGAS uses the blocks in the pool whenever it receives a write request. If all the blocks are in use, the DGAS must perform the write every time there's a request. The block size is controlled by max-async-write-bsize. Recommended range: 0 to 30 blocks.                                                                                                                                                                                                                                                                    | No       | <input type="text" value="20"/>   |
| cache-block-size             | This property specifies the block size of the per-file read cache in bytes. The cache for each file is broken into equal-sized blocks, which in turn are broken into equal-sized fragments. (The cache-frags-per-block property specifies the number of fragments per block.) A large block size improves performance by reducing the number of times the DGAS must open and close files. NOTE: The cache-block-size and cache-frags-per-block properties act together to affect cache behavior. When you change one property, consider how the change will affect the other. Recommended range: 10240 to 209715200 bytes (10 MB to 200 MB) | No       | 104857600                         |
| cache-enabled                | This property specifies whether the DGAS uses a read cache. If the value is 'True', the                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | No       | <input type="text" value="true"/> |

- Step 4** In the Value column, change values for any properties you wish.

**Note** If this DGAS will have NFS version 2 clients, you might need to set the unix-file-mode-semantics property to false. See the property description in the UI for details.

- Step 5** At the bottom of the screen, check the Force reload of DGAS log properties box if you want the DGAS to read its log properties file.

- Step 6** Click **Submit** to apply the new property values you have entered. The system displays a list of all the properties and their current values.



## Restarting the DGAS

If you reset any nondynamic properties in the previous procedure, you must shut down and restart the DGAS to make the changes take effect.

If your DGAS is registered as a service (for auto-restart), use the tools provided by your operating system to stop and restart the DGAS.

If your DGAS is *not* registered as a service (that is, if you used the **dgas --start** command rather than the **dgas --register** command to start it), follow these steps to stop it and restart it.

- Step 1** Log in to the DGAS machine as the user under which the DGAS will run.
- Step 2** Change to the Avaki installation directory. (In Unix, this defaults to /root/AvakiData-Grid<release-number>; in Windows to <system\_drive\_root>\AvakiData-Grid<release-number> (for example, C:\AvakiDataGrid70).)

- Step 3** To stop the DGAS, enter a command of this form:

```
dgas --stop [--name=<dgas-name>]
```

You must specify a <dgas-name> only if there is more than one DGAS configured on this machine.

- Step 4** To restart the DGAS, use the **dgas --start** command; the syntax is shown here. For details on the options, see [page 73](#).

```
dgas --start [--name=<server-name>] [--db-path=<local-path>]
 [--cache-path=<local-path>] [--port=<connect-port>]
```

The DGAS takes several minutes to start. Before you proceed, check the dgas.log file in the Avaki install directory — look for an INFO entry reporting “Waiting for connection” or “DGAS is in service.”

## Setting admission policies

Admission policies control which NFS and CIFS clients and networks can access data grid data through this DGAS.

### Admission policy rules

The admission policy table for each DGAS contains an ordered list of rules. The DGAS decides to permit or deny access by a client machine by matching the client's address with the policy rules. The DGAS tries to match the client address (either host or network) with each rule, starting at the top of the list. If the DGAS finds a deny rule that matches the client address, it immediately stops processing the list and blocks the client's access. Any deny rules later in the table and all allow rules are ignored.

If it finds no deny rules that match the client address, the DGAS takes no action until it has checked all the rules in the list. If the DGAS finds more than one matching allow rule, it acts on the *last* matching rule in the list—later rules supersede earlier ones.

If no match is found in the rules but a default rule is defined, the client is admitted. If no match is found and no default is defined, the client is denied access.

Keep the following in mind:

- The DGAS acts on the *first* matching deny rule in the table, if any, or
- The DGAS acts on the *last* matching allow rule in the table.
- Deny rules always override allow rules. This lets you temporarily block access from a particular client or network by adding a deny rule and leaving allow rules in place for later use.
- Default rules are used only when no allow or deny rules match.
- Once a client machine is permitted access through the DGAS, user access to data and grid tools is still controlled by ACLs on individual files and directories.

### LDAP authentication

The DGAS supports pass-through authentication for LDAP authentication services. (That is, the DGAS forwards user names and passwords to a specified Windows domain controller for authentication.) When you configure an allow rule or a default rule, you can associate it with an LDAP authentication service, for which you must provide Windows domain information.

If you do not specify a Windows domain for an allow or default rule, the DGAS uses the LDAP server for its own grid domain's LDAP authentication service in place of the Windows domain controller.

## Configuring admission policies

Follow these steps to set DGAS admission policies.

**Step 1** Log in to Avaki as a member of the Administrators group.

**Step 2** Navigate to the View DGASes screen:

Home > Server management > View DGASes

| View DGASes                                          |                           |                       |                          |                        |                         |                            |                          |
|------------------------------------------------------|---------------------------|-----------------------|--------------------------|------------------------|-------------------------|----------------------------|--------------------------|
| You are viewing the DGASes in the local grid domain. |                           |                       |                          |                        |                         |                            |                          |
| Server Name                                          | Configure                 | Cache                 | Mappings                 | Policy                 | Destroy                 | Attributes                 | Security                 |
| dgas1                                                | <a href="#">Configure</a> | <a href="#">Cache</a> | <a href="#">Mappings</a> | <a href="#">Policy</a> | <a href="#">Destroy</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="button" value="Done"/>                  |                           |                       |                          |                        |                         |                            |                          |

**Step 3** Click the Policy link for the DGAS you want to configure. The system displays the View DGAS Admission Policy screen:

| View DGAS Admission Policy                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |        |         |           |                    |                |                           |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|---------|-----------|--------------------|----------------|---------------------------|
| DGAS: dgas1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |        |         |           |                    |                |                           |
| <p>The DGAS decides to permit or deny access for a client machine by matching the client's address with the ordered list of rules below. The DGAS first tries to match the address with each deny rule, starting from the top of the list. If it finds a match, it immediately stops checking and denies access to the client. If no deny rules match the client's address, the DGAS checks for allow rules. In this case, the <i>lowest</i> rule on the list that matches the client's address takes precedence. If no rules match the client's address and a default rule is defined, then the DGAS allows access, otherwise the DGAS denies access.</p> <p>To add a new rule, specify whether to allow or deny access, and provide the IP address of the client or network in dotted decimal format (for example, 192.37.22.11 for a client or 192.37.22.0/24 for a network). For allow rules associated with an LDAP authentication service, specify the Windows domain name for the LDAP and provide the name of the primary Windows domain controller for that domain.</p> |        |         |           |                    |                |                           |
| Select                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | Number | Type    | Client IP | Auth Service       | Windows Domain | Primary Domain Controller |
| <input checked="" type="radio"/>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | 0      | default | -         | -                  |                |                           |
| <b>New Policy:</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |        | ALLOW   | . . . /   | DefaultAuthService |                |                           |
| <input checked="" type="radio"/> Replace selected row with new row<br><input type="radio"/> Insert new row above selected row<br><input type="radio"/> Delete selected row<br><input type="radio"/> Delete all rows                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |        |         |           |                    |                |                           |
| <input type="button" value="Submit"/> <input type="button" value="Cancel"/>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |        |         |           |                    |                |                           |

- Step 4** Use the New Policy row to create policy rules. First use the pull-down in the Type column to select the desired rule type: allow, deny, or default.
- Step 5** In the Client IP column, enter the IP address of the client to which the rule applies.
- Step 6** In the Auth Service column, select an authentication service for the rule. (Deny rules don't have authentication services.)
- Step 7** If you are creating an allow rule or a default rule and you want the rule to use a particular Windows domain for LDAP authentication, enter the Windows domain name in the Windows Domain column and the name of the domain controller for that domain in the Primary Domain Controller column.
- Step 8** From the radio buttons, select the appropriate option for placement of the new rule in the table. Remember that placement is important because rules are interpreted in order.
- Step 9** Click **Submit** to create the rule. The system redisplay the View DGAS Admission Policy screen showing the new rule.

To remove rules from the table, use the Delete selected row button or the Delete all rows button.

## Setting per-DGAS user and group mappings

Two sets of user and group mappings affect DGAS behavior:

- Domain-wide mappings, which are discussed in the section [“Setting domain-wide user and group mappings”](#) on page 68.
- DGAS-specific mappings, discussed here, override domain-wide mappings. Set up DGAS-specific mappings if you don't want to (or can't) set up domain-wide mappings, or if this DGAS must support dynamic user mapping so that it can serve data to Avaki Compute Grid jobs.

For details on how mappings work, see [Appendix B, “DGAS access effects”](#).

Follow these steps to set user and group mappings to be used by this DGAS only.

**Step 1** Log in to Avaki as a member of the Administrators group.

**Step 2** Navigate to the View DGASes screen:

Home > Server management > View DGASes

| View DGASes                                          |                           |                       |                          |                        |                         |                            |                          |
|------------------------------------------------------|---------------------------|-----------------------|--------------------------|------------------------|-------------------------|----------------------------|--------------------------|
| You are viewing the DGASes in the local grid domain. |                           |                       |                          |                        |                         |                            |                          |
| Server Name                                          | Configure                 | Cache                 | Mappings                 | Policy                 | Destroy                 | Attributes                 | Security                 |
| dgas1                                                | <a href="#">Configure</a> | <a href="#">Cache</a> | <a href="#">Mappings</a> | <a href="#">Policy</a> | <a href="#">Destroy</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="button" value="Done"/>                  |                           |                       |                          |                        |                         |                            |                          |

**Step 3** Click the Mappings link for the DGAS you want to configure. The system displays the View DGAS Mappings screen:

| View DGAS Mappings                                                            |         |                    |                                            |             |              |                                     |      |         |  |
|-------------------------------------------------------------------------------|---------|--------------------|--------------------------------------------|-------------|--------------|-------------------------------------|------|---------|--|
| You are viewing the user and group mappings for the DGAS named <b>dgas1</b> . |         |                    |                                            |             |              |                                     |      |         |  |
| Select                                                                        | UID/GID | Client IP/Hostname | Name                                       | Grid Domain | Auth Service | Auth Service Type                   | Type | Expires |  |
| <input type="button" value="Delete All Checked"/>                             |         |                    | <input type="button" value="Add Mapping"/> |             |              | <input type="button" value="Done"/> |      |         |  |

**Step 4** Click **Add Mapping**. The system displays the Add DGAS Mapping screen:

| Add DGAS Mapping                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |               |                                       |                             |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|---------------------------------------|-----------------------------|
| DGAS: dgas1                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |               |                                       |                             |
| <a href="#">Show users</a> <a href="#">Show groups</a>                                                                                                                                                                                                                                                                                                                                                                                                                              |               |                                       |                             |
| Select the user or group to map.                                                                                                                                                                                                                                                                                                                                                                                                                                                    |               |                                       |                             |
| All                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |               |                                       |                             |
| <a href="#">A</a> <a href="#">B</a> <a href="#">C</a> <a href="#">D</a> <a href="#">E</a> <a href="#">F</a> <a href="#">G</a> <a href="#">H</a> <a href="#">I</a> <a href="#">J</a> <a href="#">K</a> <a href="#">L</a> <a href="#">M</a> <a href="#">N</a> <a href="#">O</a> <a href="#">P</a> <a href="#">Q</a> <a href="#">R</a> <a href="#">S</a> <a href="#">T</a> <a href="#">U</a> <a href="#">V</a> <a href="#">W</a> <a href="#">X</a> <a href="#">Y</a> <a href="#">Z</a> |               |                                       |                             |
| Select                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | Name          | Authentication Service                | Authentication Service Type |
| <input type="radio"/>                                                                                                                                                                                                                                                                                                                                                                                                                                                               | Administrator | DefaultAuthService                    | Grid                        |
| <input type="radio"/>                                                                                                                                                                                                                                                                                                                                                                                                                                                               | MessagingUser | DefaultAuthService                    | Grid                        |
| <input type="radio"/>                                                                                                                                                                                                                                                                                                                                                                                                                                                               | barney        | DefaultAuthService                    | Grid                        |
| <input type="radio"/>                                                                                                                                                                                                                                                                                                                                                                                                                                                               | betty         | DefaultAuthService                    | Grid                        |
| <input type="radio"/>                                                                                                                                                                                                                                                                                                                                                                                                                                                               | fred          | DefaultAuthService                    | Grid                        |
| <input type="radio"/>                                                                                                                                                                                                                                                                                                                                                                                                                                                               | wilma         | DefaultAuthService                    | Grid                        |
| <input type="button" value="Continue"/>                                                                                                                                                                                                                                                                                                                                                                                                                                             |               | <input type="button" value="Cancel"/> |                             |

- Step 5** To display a list of users, click **Show users** (top of screen). To display a list of groups, click **Show groups**.
- Step 6** Click in the Select column to choose a user or group to map.
- Step 7** Click **Continue**. The system displays a new screen:

**Add DGAS Mapping**

Enter up to five UIDs or GIDs to be mapped to **barney**. For each UID/GID, specify the client IP or hostname from which requests are expected. You may enter the same UID/GID more than once, specifying a different client IP or hostname for each entry. When entering an IP, you may *not* specify a netmask.

| UID/GID              | Client IP/Hostname   |
|----------------------|----------------------|
| <input type="text"/> | <input type="text"/> |
| <input type="text"/> | <input type="text"/> |
| <input type="text"/> | <input type="text"/> |
| <input type="text"/> | <input type="text"/> |
| <input type="text"/> | <input type="text"/> |

- Step 8** Enter a UID or GID and an associated IP address or DNS name for each user account or group that you are mapping.
- Step 9** Click **Submit**. A list of mappings for the DGAS appears.

## Configuring a cache service for a DGAS

By default, a DGAS uses only its own internal caches. You can configure a DGAS to also use a cache service on another grid server. For example, you might choose to use the cache service on a grid server near your DGAS if the source data is at a remote location. In such a case, users who access the remote data through the DGAS might have access to cached copies even when the network link to the remote data source is down. (We refer to this as *remote caching* because the cache is remote from the data source.)

Follow these steps to set the DGAS to use a cache service on another Avaki server.

**Step 1** Log in to Avaki as a member of the Administrators group.

**Step 2** Navigate to the View DGASes screen:

Home > Server management > View DGASes

### View DGASes

You are viewing the DGASes in the local grid domain.

| Server Name | Configure                 | Cache                 | Mappings                 | Policy                 | Destroy                 | Attributes                 | Security                 |
|-------------|---------------------------|-----------------------|--------------------------|------------------------|-------------------------|----------------------------|--------------------------|
| dgas1       | <a href="#">Configure</a> | <a href="#">Cache</a> | <a href="#">Mappings</a> | <a href="#">Policy</a> | <a href="#">Destroy</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

**Step 3** Click the Cache link for the DGAS whose cache you want to configure. The system displays the Manage DGAS Cache screen:

### Manage DGAS Cache

DGAS: dgas1

#### Manage Internal Cache Service

- [Clear credentials](#) Clear all cached user credentials
- [Delete from cache](#) Delete a grid directory or file from DGAS cache
- [Save cache](#) Save a copy of the cache for a DGAS
- [Sync file with proxy cache](#) Force a DGAS to fetch grid directories or files from proxy cache if the cached version is newer than the version on the DGAS; the DGAS will copy the directory or file if it is not cached
- [View/modify cache statistics](#) View and reset all statistics counters for the DGAS cache

#### Manage External Cache Service

- [Set external cache service](#) Associate an external proxy cache service with a DGAS

**Step 4** Click **Set external cache service**. The system displays the Set External Cache Service screen.

**Step 5** In the New cache service pull-down, select the name of the Avaki server whose cache service you want the DGAS to use.

**Step 6** Click **Submit**.

---

## Setting up NFS clients

The following sections describe how to set up an NFS client:

- [“System requirements for NFS clients,”](#) below
- [“NFS-mounting Avaki directories”](#) on page 92

### System requirements for NFS clients

Each computer on which you wish to mount Avaki directories must run an NFS client. Many implementations of Unix include an NFS client, but most versions of Windows do not. If necessary, install NFS client software on each client machine. Avaki supports NFS versions 2 and 3. There are no restrictions on the operating system.

The NFS client machine and the DGAS machine may be the same.

### NFS-mounting Avaki directories

Log in to the client machine. Following the instructions provided for your NFS client software, mount the Avaki directories from the data catalog.

- If the client machine is running Unix, you must log in as root to perform the NFS mount.
- If the client machine is running any operating system other than Unix, you must log in as a user with privileges to mount a file system.

Remember that the mounting user must be mapped to a grid user account. Setting up domain-wide user mappings is described in [“Setting domain-wide user and group mappings”](#) on page 68; setting mappings for the current DGAS only is describe in [“Setting per-DGAS user and group mappings”](#) on page 88.

---

**Note** Even after you have set up a mapping for the user doing the mount (or a default mapping), you might have to wait a few minutes before you can mount Avaki directories. This is because the DGAS caches credentials, and the cache must be updated with the new credentials from the mapping.

Items to include in the NFS **mount** command:

- If you turned off the port mapper update feature on the Connect DGAS screen ([page 82](#)), include the data grid access server’s NFS port number (which is 2049 by default, but can also be changed on the Connect DGAS screen).



- An IP address or DNS name for the DGAS machine. Do not use “localhost” in the mount command because it might cause the mount to fail. localhost often resolves to 127.0.0.1, which might not be included in your DGAS admission policies.

## Examples

This command mounts the whole Avaki file system in the local directory /avakifs. (Replace “thishost” with the name or IP address of the machine.)

```
$ mount -t nfs thishost:/ /avakifs
```

---

**Note** In the **mount** command, when you specify a machine that has more than one IP address, it is best to use an IP address rather than a machine name. This is because the name may resolve to any of the machine’s IP addresses, and some of them may not be included in domains defined in the data grid access server configuration files.

This command mounts the Avaki file system under /home/admin/docs-share to the local directory /home/docs, from a server running on 162.34.10.1.

```
$ mount -t nfs 162.34.10.1:/home/admin/docs-share /home/docs
```

This command also mounts the Avaki file system under /home/admin/docs-share to the local directory /home/docs, from a server running on 162.34.10.1. In this case, because the client is using Windows Services for Unix, we use the “-o nolock” option to turn off file locking.

```
$ mount -t nfs -o nolock 162.34.10.1:/home/admin/docs-share /home/docs
```

This command uses an NFS URL to mount the top level grid directory, /, to the local directory /mnt/avaki. The Avaki directory is mounted from a data grid access server running on 172.21.33.1, port 2000. You can use NFS URLs only on NFS clients that support them, such as Solaris. (If you want to mount a grid directory other than /, don’t use an NFS URL—see the mounting instructions for your NFS client.)

```
% mount nfs://172.21.33.1:2000/ /mnt/avaki
```

---

## Setting up CIFS clients

A CIFS client is a machine running Windows 2000 that accesses data catalog files and directories through a CIFS share that has been configured in the data grid. The CIFS client does not require any Avaki software.

Before you can access a CIFS share from a client machine, you must configure the share in the data grid. See [“Creating and managing CIFS shares” on page 125](#) for instructions.

CIFS shares are exported to the Windows network neighborhood. To access a CIFS share from a CIFS client, map the share as a network drive. For instructions, see [“Accessing CIFS shares” on page 204](#).

---

## Setting up proxy servers and routing tables

Before you can interconnect two Avaki domains, you must ensure that they share a uniform IP space—that you can ping from one to the other. If you can’t—for example, because the Avaki domains are separated by a firewall or Network Address Translation (NAT)—the following tasks must be performed by administrators in the two Avaki domains:

- Set up an Avaki proxy server in the data provider’s Avaki domain.
- Enter the provider’s proxy details in the consumer domain’s proxy routing table.

For a one-way interconnect, perform each task once. In a two-way interconnect, each grid domain is both a provider and a consumer, so you need to perform each task for each domain.

For instructions on setting up routing tables, see [“Connecting to a provider domain” on page 292](#). For instructions on configuring proxy servers, see [“Enabling interconnection access” on page 299](#).

## Setting up command clients

A command client is used only as a terminal into the Avaki domain; it does not contribute resources to the domain. It allows you to issue Avaki commands and access grid data. There can be many command clients within an Avaki domain.

Set up command clients after the grid domain controller has been configured.

A command client can be stand-alone—that is, the only Avaki software on its machine—or it can run in tandem with a grid server. You must connect a command client before you can issue commands on a grid server.

To set up an Avaki command client, do the following:

- Step 1** Run the appropriate Avaki installer on the target machine, and select a command-line client installation. (That is, deselect all other components.) For instructions, see [“Installing in Unix” on page 16](#) or [“Installing in Windows” on page 22](#). (This step isn’t necessary on a machine where a grid server is installed.)
- Step 2** To start a command client, open a command window and enter a command of this form:
- ```
avaki client --connect <GDC-machine>
```

where <GDC-machine> is the name or IP address of the machine that serves as the grid domain controller for this grid domain. For example:

```
$ avaki client --connect Flintstone
```

Note If your GDC is not using the default connect port, 3099, you must enter the GDC’s connect port number after the name or IP address, like so:

```
$ avaki client --connect Flintstone:221122
```

- Step 3** Log in to the Avaki domain. In this example, user Fred enters a password at the Password prompt. At the next two prompts Fred presses **Enter** to accept the defaults (DefaultAuthService and Grid service type):

```
$ avaki login Fred
Password:
Auth service [DefaultAuthService]:
Service type [(Grid), Nis, Ldap]:
Fred logged in successfully
$
```

- Step 4** Avaki cache services can store files or results produced by remote services (database operations or data services) so that the data does not have to be fetched over a long distance to satisfy every new request from a client. (This is called remote caching.) To associate a cache service on a nearby grid server with this instance of the command client, use this command:

```
avaki cache --set --cli <cache-service-path>
```

where <cache-service-path> is the grid path to the cache service. For example:

```
$ avaki cache --set --cli  
/System/LocalDomain/Services/CacheServices/bedrock.avaki.com
```

Managing Avaki servers

This chapter covers the following topics:

- “Categories,” below
- “Displaying domain information” on page 98
- “Finding your version number” on page 99
- “Monitoring” on page 99
- “Logging” on page 107
- “Configuring execution services” on page 109
- “Managing failover” on page 111
- “Backing up and restoring Avaki servers” on page 113
- “Managing DGAS caches” on page 117
- “Creating and managing CIFS shares” on page 125
- “Setting system properties” on page 128
- “System property reference” on page 131

Categories

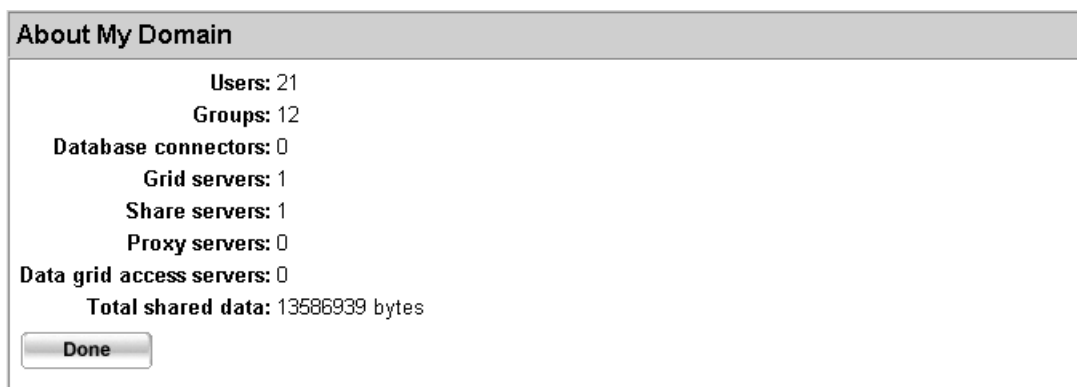
The data catalog provides a hierarchy parallel to the grid directory structure called *categories*. You can use categories to classify the contents of your data catalog. For more information, see [“Using and managing categories” on page 221](#).

Displaying domain information

To help you track usage of Avaki software, the web user interface provides an information screen that lists, for the current Avaki domain, the number of user accounts, number of groups, number of database connectors, number of each type of Avaki server, and amount of data shared into the grid.

To display information about your Avaki domain, do the following:

- Step 1** Log into the Avaki domain as a member of the Administrators group.
- Step 2** On the Welcome screen of the web UI, click the **About my domain** link in the upper right corner. The About My Domain screen appears.



- Step 3** Click **Done** to return to the Welcome screen.

Finding your version number

To display the version number of the Avaki software on a machine, do one of the following:

- Display the VERSION file in the Avaki installation directory.
- At a command prompt in the Avaki installation directory, enter **avaki help**. Look for the version in the first line of the resulting output.

Monitoring

Avaki software offers two forms of monitoring:

- Operations status monitoring lets you display the status of data services, database operations, share rehashes, and view generators running on a specified grid server. See [“Viewing the operations status screen,”](#) below.
- Monitor services enable you to configure GDCs and grid servers to monitor other Avaki servers in the same grid domain. See [“Using monitor services” on page 100](#).

Viewing the operations status screen

Navigate to the Grid Server Operations Status screen:

Home > Services management > View grid server operations status

Grid Server Operations Status

Server:

Choose the server whose operations status you want to view.

In the Server field, select a grid server from the pull-down menu. The system displays a list of current and recent operations on this grid server and their start and end times. (By default, recent operations are those completed within the last 15 minutes. To

change the expiration interval for operations, see [“Setting the operations status expiration interval” on page 145.](#))

Grid Server Operations Status

Server:

Report generated at September 27, 2004 1:15:30 PM EDT

Type	Name	User	Start Time	End Time
DatabaseOperation	CherylDomain.MyDBCConnector.MyDBOperation	Administrator	September 27, 2004 1:11:06 PM EDT	September 27, 2004 1:11:06 PM EDT
ShareRehash	MyAvakiShare_1095956943961	Administrator	September 27, 2004 1:11:04 PM EDT	September 27, 2004 1:11:05 PM EDT
DatabaseOperation	CherylDomain.MyDBCConnector.MyDBOperation	Administrator	September 27, 2004 1:06:04 PM EDT	September 27, 2004 1:06:05 PM EDT
ShareRehash	MyAvakiShare_1095956943961	Administrator	September 27, 2004 1:06:03 PM EDT	September 27, 2004 1:06:04 PM EDT
DatabaseOperation	CherylDomain.MyDBCConnector.MyDBOperation	Administrator	September 27, 2004 1:01:03 PM EDT	September 27, 2004 1:01:04 PM EDT
ShareRehash	MyAvakiShare_1095956943961	Administrator	September 27, 2004 1:01:02 PM EDT	September 27, 2004 1:01:03 PM EDT

Click or mouse over a name in the User column to get information about that user’s authentication service.

The Grid Server Operations Status screen updates automatically every 5 minutes. You can also click the **Refresh** button to update it.

Using monitor services

You can configure a monitor service on a GDC or other grid server to monitor another server, including a primary or secondary GDC, grid server, share server, data grid access server, or proxy server. Monitoring results are stored, along with other grid events, in a log4j log. You can use the log to generate alerts about monitoring failures and other conditions. For more on logging monitoring results, see [“Server logs” on page 107.](#)

To enable monitoring, you create a monitor service for each of the servers you want to monitor, then set up tests in each monitor service. There are two types of test:

- Ping tests measure the time it takes the target server to respond to ICMP echo request messages. Ping tests check network connectivity to and from the target server and tell you whether the server is running. Ping tests are faster than message tests.

- Message tests check the asynchronous messaging system that sends update notifications for generated views and updates dependencies for data services and database operations. Message tests send JMS messages to a target server, listen for a response message, then measure the round-trip time.

The monitor service runs the tests continuously (at intervals you specify), logs response times and failures, and calculates a running average of response times.

Note To monitor a GDC or a server, you must set up a monitor service on a separate server—a GDC or server cannot effectively monitor itself.

The procedures in this section cover these topics:

- [“Setting up a monitor service,”](#) below
- [“Setting up monitor tests”](#) on page 102
- [“Viewing test results”](#) on page 103
- [“Disabling and re-enabling monitor tests”](#) on page 104
- [“Stopping, restarting, and deleting monitor tests”](#) on page 105

Logs are covered in the section that follows this one: [“Logging”](#) on page 107.

Setting up a monitor service

A monitor service manages the tests that one grid server or GDC uses to monitor another grid server. You must set up a monitor service for each server you want to monitor. Follow these steps to set up a monitor service:

Step 1 Log in to the grid domain as a member of the Administrators group.

Step 2 Navigate to the Create Monitor Service screen:

Home > Service management > Create monitoring service

Create Monitor Service

A monitor service manages the tests that one grid server or grid domain controller (GDC) uses to monitor another grid object. To create a monitor service, specify a name for the monitor service, and then select the grid server or GDC on which the new monitor service will run.

Service name:

Grid server:

Step 3 Fill in the form:

- Service name: Enter a name for the monitor service.
- Grid server: From the pull-down list, select the grid server or GDC on which this new monitor service will run.

Step 4 Click **Submit**. The system creates the monitor service.

Setting up monitor tests

You must set up tests to monitor a specific primary or secondary GDC, grid server, share server, data grid access server (DGAS), or proxy server in your grid domain. Follow these steps.

Step 1 Log in to the grid domain as a member of the Administrators group.

Step 2 Navigate to the View Monitoring Services screen:

Home > Service management > View monitoring services

View Monitoring Services					
Service Name	Configure Service	View Reports	Delete Service	Attributes	Security
monitor1	Configure	View	Delete	Attributes	Security

Step 3 Click **Configure** for the monitor service to which you want to add tests. On the screen that appears, scroll down to the Add a Server Monitoring Test area.

Add a Server Monitoring Test	
To add a monitor test for a server, specify the action, the target server, the number of seconds to wait for a test to complete and the number of seconds between tests.	
Action: <input type="text" value="Ping"/>	Specify the action to be performed as part of this test.
Target Path: <input type="text"/>	select
Timeout: <input type="text" value="1"/>	Specify the duration in seconds to wait for a test to complete before marking the test a failure.
Frequency: <input type="text" value="60"/>	Specify the number of seconds between consecutive tests.
<input type="button" value="Submit"/>	<input type="button" value="Cancel"/>

Step 4 Fill in the form:

- **Action:** Use the pull-down menu to select the type of test you want to add, Ping or Message. (For descriptions, see [“Using monitor services” on page 100.](#))
- **Target path:** To fill in this field, click **Select**. Choose the type of server to monitor from the Target type pull-down, then select a server from the list. When you click **Submit**, the path to the server you selected is added to the Target path field in the Add a Server Monitoring Test area.

Target type:

Select Monitor Target	
Select	Target Name
<input type="radio"/>	Antimony.avaki.local
<input type="radio"/>	localhost

- **Timeout:** Specify, in seconds, how long the monitor service should wait for this test to complete. The test fails when it does not finish within the timeout period.
- **Frequency:** Specify, in seconds, how long the monitor service should wait between tests.

Step 5 Click **Submit** to create the test.

Viewing test results

Follow these steps to display reports on a monitor service’s tests.

Step 1 Log in to the grid domain as a member of the Administrators group.

Step 2 Navigate to the View Monitoring Services screen:

Home > Service management > View monitoring services

View Monitoring Services					
Service Name	Configure Service	View Reports	Delete Service	Attributes	Security
monitor1	Configure	View	Delete	Attributes	Security

Step 3 Click **View** for the monitor service whose test results you want to display.

- Step 4** In the View Monitor Reports screen that appears, go to the View Monitor Tests area and select a test.

View Monitor Reports

Monitor reports show the status of a server for which a monitor test has been added. To add a monitor test to a monitoring service, go to the [View Monitoring Service](#) screen and click the **Configure** link beside the monitoring service to modify, then provide the information requested in the "Add a Server Monitoring Test" section.

Monitor Report	
Target:	/System/Domains/Bedrock/Servers/Antimony.avaki.local
Status:	Ping successful.
Last status change time:	Thu Sep 30 15:18:03 EDT 2004
Time of last attempt:	Fri Oct 01 13:40:03 EDT 2004
Average time:	2

Monitor Report	
Target:	/System/Domains/Bedrock/Servers/Antimony.avaki.local
Status:	Message successful.
Last status change time:	Thu Sep 30 15:19:04 EDT 2004
Time of last attempt:	Fri Oct 01 13:39:04 EDT 2004
Average time:	453

- Step 5** Click **Done** to return to the View Monitoring Services screen.

Disabling and re-enabling monitor tests

Follow these steps to disable or re-enable all the tests in a monitor service. (If you want to stop a particular test, see [“Stopping, restarting, and deleting monitor tests”](#) on page 105.)

- Step 1** Log in to the grid domain as a member of the Administrators group.

- Step 2** Navigate to the View Monitoring Services screen:

Home > Service management > View monitoring services

View Monitoring Services

Service Name	Configure Service	View Reports	Delete Service	Attributes	Security
monitor1	Configure	View	Delete	Attributes	Security

- Step 3** Click **Configure** for the monitor service whose tests you want to disable.

- Step 4** In the screen that appears, go to the Manage Active Monitor Tests area and click **Stop**.

Manage Active Monitor Tests

Click **Stop** to temporarily stop all active monitoring tests or click **Restart** to restart all inactive tests. To delete all active monitoring tests, click **Delete**.

To resume the tests, click **Restart**.

Stopping, restarting, and deleting monitor tests

Follow these steps to stop, restart, or delete individual tests in a monitor service.

- Step 1** Log in to the grid domain as a member of the Administrators group.

- Step 2** Navigate to the View Monitoring Services screen:

Home > Service management > View monitoring services

View Monitoring Services

Service Name	Configure Service	View Reports	Delete Service	Attributes	Security
monitor1	Configure	View	Delete	Attributes	Security

- Step 3** Click **Configure** for the monitor service whose tests you want to stop, restart, or delete.

- Step 4** In the screen that appears, go to the View Monitor Tests area and select a test.

View Monitor Tests

The following monitor tests are present:

	Action	Target	Timeout	Frequency	Active?
<input type="radio"/>	MessageTask	/System/Domains/Bedrock/Servers/Antimony.avaki.local	1	600	true
<input type="radio"/>	PingTask	/System/Domains/Bedrock/Servers/Antimony.avaki.local	1	60	true

- Step 5** Click **Stop** to turn off the selected test, **Restart** to turn it on again, or **Delete** to remove the selected test from the monitor service.

Deleting monitor services

Follow these steps to remove a monitor service and all its tests.

Step 1 Log in to the grid domain as a member of the Administrators group.

Step 2 Navigate to the View Monitoring Services screen:

Home > Service management > View monitoring services

View Monitoring Services					
Service Name	Configure Service	View Reports	Delete Service	Attributes	Security
monitor1	Configure	View	Delete	Attributes	Security
<input type="button" value="Done"/>					

Step 3 Click **Delete** for the monitor service you want to remove. The system removes the monitor service and redisplay the View Monitoring Services screen.

Logging

This section discusses three types of logs:

- “[Server logs](#),” below
- “[Request logs](#)” on page 108
- “[Audit logs](#)” on page 108

Server logs

Server logs, which are maintained by log4j, record grid events and monitoring results. By default, the monitoring results that server logs record include only the failure of monitoring tests, but you can configure log4j to record successful tests too.

Each Avaki server maintains a server log file, as does Avaki Studio. The log file for each component resides in the Avaki installation directory:

```
<Avaki_install_dir>/jboss/server/grid-server/log/server.log
<Avaki_install_dir>/jboss/server/proxy-server/log/server.log
<Avaki_install_dir>/jboss/server/share-server/log/server.log
<Avaki_install_dir>/dgas.log
<Avaki_install_dir>/avaki_studio/log/studio.log
```

Monitoring results are indicated by these entries in the log files:

Ping failed.	Indicates that a ping test failed.
Message failed.	Indicates that a message test failed.

If you have configured log4j to log monitor test successes, you’ll see these entries:

Ping successful.	Indicates that a ping test passed.
Message successful.	Indicates that a message test passed.

For information on configuring server logs, see [Appendix A, “Setting up log properties files”](#).

Request logs

Request logs record HTTP requests. These Avaki servers maintain request logs:

- Each proxy server keeps a request log for proxy traffic.
- Each grid server keeps a request log for UI traffic.

The request log files can be found in the Avaki installation directory:

```
<Avaki_install_dir>/jboss/server/proxy-server/log/<date>.request.log
```

```
<Avaki_install_dir>/jboss/server/grid-server/log/<date>.request.log
```

For more information on configuring request logs, see [Appendix A, “Setting up log properties files”](#).

Audit logs

Audit logs record grid events that you specify (for example, the creation or execution of a database operation). The events are logged in a common store (such as a file or database) that you also specify.

Audit logs are controlled by log4j and you configure them in the log properties file for each Avaki component. For more information, see [“Server log properties files” on page 317](#). You can use the AUDITFILE appender provided in the log properties file, modify the AUDITFILE appender, or write custom appenders to capture the events you want to audit and send them to a common store.

The default locations of audit logs for Avaki servers are as follows. Audit logs will be created only if you enable audit logging.

```
<Avaki_install_dir>/jboss/server/grid-server/log/audit.log
```

```
<Avaki_install_dir>/jboss/server/proxy-server/log/audit.log
```

```
<Avaki_install_dir>/jboss/server/share-server/log/audit.log
```

```
<Avaki_install_dir>/dgas_audit.log
```

Configuring execution services

Running on every grid server is an *execution service* that contains the logic for running the data services associated with its grid server. Execution services enable remote grid servers to transparently access data and data services. (For more on data services, see *Data Integration with Sybase Avaki Studio* and the *Sybase Avaki EII Provisioning and Advanced Data Integration Guide*.)

There are two features you can configure for execution services:

- **Pooling execution services**
You can pool or cluster execution services by associating several execution services (each on its own grid server) with a single host grid server. A data service on the pool's host grid server may be run by any execution service in the pool—or on multiple grid servers simultaneously. When you're using a execution service, it's not possible to specify which grid server in the pool will run a particular data service, or to prevent a data service from running on certain grid servers.
- **Maximum concurrent data services**
By default, an execution service can run an unlimited number of data services at once. You might want to configure a smaller number. Consider these factors:
 - How powerful is the processor on the execution service machine?
 - How complex are the data services that will be run by this execution service?
 - Is this execution service the only one available on this grid server, or is it part of a pool?
 - How frequently are the data services likely to be run?
 - Is this execution service machine dedicated to executing data services, or must it reserve cycles for other tasks?

If you limit the number of data services that can run concurrently, data services in excess of the limit are queued until the execution service can accommodate them without exceeding the limit.

Caution If you have a chain of dependent data services—data service A gets input from data service B, which gets input from data service C, and so on—the execution service will lock up if the number of concurrent data services is set to a value lower than the number of data services in the chain. This happens because an attempt to run any data service causes the execution service to run all the data services that provide input (for example, running data service C causes A and B to run as well).

Note The maximum concurrent data services setting interacts with a system property that controls simultaneous reads of data service results and database operation results from a cache service. Such reads are performed by execution services. Your execution services, particularly if they are pooled, will not perform efficiently unless the system property is set to an appropriate value. For details on the system property, `com.avaki.cache.maxReaderThreads`, see “[Configuring read threads for cache services](#)” on page 131.

Follow these steps to configure an execution service.

Step 1 Log in to the grid domain as a member of the Administrators group.

Step 2 Navigate to the View Execution Services screen:

Home > Service management > View execution services

View Execution Services

You are viewing the execution services in the local grid domain.

Name	View/Edit	Concurrent Operations	Pooled With
Antimony.avaki.local	View/Edit	5	<i>Not pooled with any other servers.</i>
Antimony.avaki.local_Port_11099	View/Edit	5	<i>Not pooled with any other servers.</i>
Antimony.avaki.local_Port_21099	View/Edit	5	<i>Not pooled with any other servers.</i>

- Step 3** Click **View/Edit** for the grid server whose execution service you want to configure. (If you want to set up a pool, choose the grid server that will host the pool.) The View/Edit Execution Service screen appears.

- Step 4** To limit the load imposed by this execution service, click the No more than button (under Maximum concurrent data services). Then enter a number in the adjacent field. When you choose a number, consider the factors listed at the beginning of this procedure.
- Step 5** To create a pool of execution services, click one or more boxes in the Pool with field.
- Step 6** Click **Submit** to save your changes. The system displays the View Execution Services page showing the changes you made.

Managing failover

If you have configured a secondary GDC in your grid domain, the secondary GDC handles requests directed to the primary GDC if the primary fails or becomes unreachable. (The secondary may also respond to requests if the primary is heavily loaded.)

Failover happens when the secondary GDC takes over from the primary.

Identifying failover

It's easiest to determine that failover to the secondary GDC has occurred if you have configured a monitor service for the primary GDC. For instructions on configuring a monitor service, see [“Monitoring” on page 99](#).

If you're not sure whether failover has occurred, try to create a new directory under /, the top-level grid directory. If you see an error that says you can't write to the secondary GDC (backup or replica), consider doing the following:

- Restart your primary GDC
- Troubleshoot the primary GDC's host machine
- Troubleshoot the network

If the fault lies in the GDC or its host, failover has occurred.

Effects of failover

Keep the following in mind when failover from the primary to the secondary occurs:

- Because the secondary GDC is read-only, you cannot perform user administration tasks on any authentication services that reside on the primary GDC. This restriction applies to adding and removing users and groups, adding users to and removing users from groups, and changing passwords. Users can, however, log in and out.
- Avaki shares that use the primary GDC's built-in share server, or that use share servers connected to the primary GDC, are unavailable until the primary returns to service. (Other Avaki shares—that is, shares that use share servers associated with other grid servers—are not affected by failover.)
- Grid directories that reside on a grid server other than the primary GDC will be available if failover occurs, but will *not* be available if their host grid server fails.
- You cannot use the Avaki web UI through the primary GDC. Point your browser to the secondary GDC or to another grid server in your grid domain:

```
http://<grid-server-or-2GDC-name>:7080
```

- You can also log in to the grid domain from the command line. You can do this from the secondary GDC, from a grid server, from a command client, or, if its machine is still up, from the primary GDC. If you're logging in from the primary, use this command first:

```
avaki client --connect <secondary-machine-name>
```

Use the **avaki login** command to log in.

Backing up and restoring Avaki servers

Each grid domain controller, grid server and share server maintains databases and internal shares that store all the persistent information needed for the portion of the grid domain supported by that server, including user account records, ACLs and attributes for all grid objects, share properties, and so on.

Read this section for instructions on backing up persistent state and restoring it if it's lost or damaged. This section covers these topics:

- “[Managing the persistent state of GDCs and grid servers](#),” below
- “[Managing the persistent state of share servers](#)” on page 115

Data grid access servers also maintain databases, but it is not necessary to back them up. Proxy servers do not maintain state databases.

Managing the persistent state of GDCs and grid servers

The procedures in this section tell you how to back up and restore persistent state on GDCs and grid servers.

The databases and internal shares that contain the persistent state of a grid server or GDC are stored in the subdirectories under `<Avaki-install-dir>/jboss/server/grid-server/db/`.

Backing up the persistent state of a grid server or a GDC

The procedure that follows shows how to obtain a snapshot zip archive of a server's persistent state by executing the **avaki backup** command manually. You can also create a script to run the **avaki backup** command. We recommend that you run backups for each grid server and GDC on a regular basis.

- Step 1** Log in to the grid server's or GDC's machine as a system administrator or as the user who installed Avaki software on that machine.
- Step 2** If you haven't previously used the Avaki command line from this machine, connect the command client to the grid domain:

```
avaki client --connect <GDC-machine>
```

where `<GDC-machine>` is the name or IP address of the machine on which this grid domain's GDC runs.

- Step 3** From the command line, log in to Avaki as a member of the Administrators group. For example:

```
c:\ avaki login Administrator
```

- Step 4** Use this command to copy the database:

```
avaki backup --snapshot [--grid-server=<grid-server>]
<filename>.zip
```

where

<grid-server> is the name of the grid server you want to back up. This is optional; if you omit it, the command backs up the GDC.

<filename>.zip is the name of the backup zip archive to be created. It can be an absolute or relative path.

- Step 5** Copy the backup archive to a safe place on another machine.

Restoring the persistent state of a grid server or a GDC

You can restore a server remotely if it's still in operation, or you can restore a downed server by logging in to its machine. This section includes both procedures.

Remotely restoring an operating server. If a grid server or GDC is operational, you can remotely restore its state from a backup archive. We strongly recommend that you perform the restoration at a time when users are not reading and writing files through the grid server or GDC, or when read/write traffic is light.

Follow these steps to restore the server remotely.

- Step 1** Log in to the grid server's or GDC's machine as a system administrator or as the user who installed Avaki software on that machine.
- Step 2** From the command line, log in to Avaki as a member of the Administrators group. For example:

```
c:\ avaki login Administrator
```

- Step 3** Use this command to restore the database:

```
avaki backup --recover <filename>.zip
```

where <filename>.zip is the local path to your most recent backup.

If the server or GDC is running, it will restart itself when the restoration is complete.

Restoring a server that's not responding. If the grid server or GDC is not operational or not responding, follow these steps.

Step 1 Use this command to shut the server down:

```
grid-server --stop [--port=<port-number>]
```

The <port-number> is unnecessary if the server is running on 3099, the default connect port.

If your share server is registered as a service, you can also use operating system tools to stop the share server. If you are unable to shut down the server using the **grid-server --stop** command, use OS tools to kill the server's process.

Step 2 Delete the contents of the server's db directory, <Avaki-install-dir>/jboss/server/grid-server/db

Step 3 Extract the contents of the backup snapshot archive into that directory. (You can use any zip file utility to perform the extraction.)

Step 4 Restart the server:

```
grid-server --start [--port=<port-number>]
```

(If your server is registered as a service, you can also use operating system tools to restart the share server.)

Managing the persistent state of share servers

The procedures in this section explain how to back up and restore persistent state on share servers.

The databases and internal shares that contain the persistent state of share servers are stored in the subdirectories in <Avaki-install-dir>/jboss/server/share-server/db/.

Backing up a share server

The procedure that follows shows how to obtain a snapshot zip archive of a share server's persistent state by executing the **avaki backup** command manually. You can also create a script to run the **avaki backup** command. We recommend that you run backups for each share server on a regular basis.

Step 1 Log in to the share server's machine as a system administrator or as the user who installed the share server.

- Step 2** To ensure that the database files are in a consistent state, shut the share server down before backing it up:

```
share-server --stop [--port=<port-number>]
```

where <port-number> is the port on which the share server is running. You need not specify a port number if the share server is running on the default port, 2099.

(If your share server is registered as a service, you can also use operating system tools to stop the share server.)

- Step 3** Copy the contents of <Avaki-install-dir>/jboss/server/share-server/db to a safe place on another machine:

- Step 4** Restart the share server:

```
share-server --start [--port=<port-number>]
```

(If your share server is registered as a service, you can also use operating system tools to restart the share server.)

Restoring a share server

To restore a share server's persistent state, you must previously have performed a backup. (See [“Backing up a share server,”](#) above, for instructions on performing backups.)

Caution This procedure requires you to shut down the share server.

Follow these steps to restore a share server.

- Step 1** If the share server is running, warn users that you are going to shut it down.
- Step 2** Log in to the share server's machine as a system administrator or as the user who installed the share server.
- Step 3** Shut down the share server:

```
share-server --stop [--port=<port-number>]
```

where <port-number> is the port on which the share server is running. You need not specify a port number if the share server is running on the default port, 2099.

If your share server is registered as a service, you can also use operating system tools to stop the share server. If you are unable to shut down the server using the **share-server --stop** command, use OS tools to kill the server's process

Step 4 Remove any existing content from the db directory,
<Avaki-install-dir>/jboss/server/share-server/db.

Step 5 Copy the backup archive into the db directory.

Step 6 Restart share server:

```
share-server --start [--port=<port-number>]
```

(If your share server is registered as a service, you can also use operating system tools to restart the share server.)

Managing DGAS caches

A DGAS maintains its own internal caches to store copies of the files and directories it serves to NFS and CIFS clients, as well as other information. The DGAS caches may get their data directly from the source file systems, or from the cache service associated with the DGAS. DGAS caches are different from cache services, which run on grid servers.

The procedures that follow explain how to manage a DGAS's internal cache:

- [“Clearing cached credentials,”](#) below
- [“Deleting files and directories from the cache”](#) on page 119
- [“Saving a copy of the cache”](#) on page 120
- [“Syncing the cache”](#) on page 121
- [“Viewing and modifying cache statistics”](#) on page 123
- [“DGAS properties and cache size”](#) on page 124

Clearing cached credentials

By default, the DGAS credentials cache, which stores information about users who have logged in recently, is flushed every two hours. If you add, delete, or update users or groups and need the changes to take effect immediately, follow these steps to clear cached user credentials from the DGAS.

Step 1 Log in to Avaki as a member of the Administrators group.

Step 2 Navigate to the View DGASes screen:

Home > Server management > View DGASes

View DGASes

You are viewing the DGASes in the local grid domain.

Server Name	Configure	Cache	Mappings	Policy	Destroy	Attributes	Security
dgas1	Configure	Cache	Mappings	Policy	Destroy	Attributes	Security

Step 3 Click the Cache link for the DGAS whose cache you want to configure. The system displays the Manage DGAS Cache screen:

Manage DGAS Cache

DGAS: dgas1

Manage Internal Cache Service

- [Clear credentials](#) Clear all cached user credentials
- [Delete from cache](#) Delete a grid directory or file from DGAS cache
- [Save cache](#) Save a copy of the cache for a DGAS
- [Sync file with proxy cache](#) Force a DGAS to fetch grid directories or files from proxy cache if the cached version is newer than the version on the DGAS; the DGAS will copy the directory or file if it is not cached
- [View/modify cache statistics](#) View and reset all statistics counters for the DGAS cache

Manage External Cache Service

- [Set external cache service](#) Associate an external proxy cache service with a DGAS

Step 4 Click **Clear credentials**. The system displays the Clear Cached User Credentials screen.

Step 5 Click **Submit** to clear credentials.

Deleting files and directories from the cache

Follow these steps to clear files or directories from the DGAS cache.

Step 1 Log in to Avaki as a member of the Administrators group.

Step 2 Navigate to the View DGASes screen:

Home > Server management > View DGASes

View DGASes

You are viewing the DGASes in the local grid domain.

Server Name	Configure	Cache	Mappings	Policy	Destroy	Attributes	Security
dgas1	Configure	Cache	Mappings	Policy	Destroy	Attributes	Security

Step 3 Click the Cache link for the DGAS whose cache you want to configure. The system displays the Manage DGAS Cache screen:

Manage DGAS Cache

DGAS: dgas1

Manage Internal Cache Service

[Clear credentials](#) Clear all cached user credentials

[Delete from cache](#) Delete a grid directory or file from DGAS cache

[Save cache](#) Save a copy of the cache for a DGAS

[Sync file with proxy cache](#) Force a DGAS to fetch grid directories or files from proxy cache if the cached version is newer than the version on the DGAS; the DGAS will copy the directory or file if it is not cached

[View/modify cache statistics](#) View and reset all statistics counters for the DGAS cache

Manage External Cache Service

[Set external cache service](#) Associate an external proxy cache service with a DGAS

Step 4 Click **Delete from cache**. The Delete File from Cache screen appears.

Step 5 In the Target path field, enter the grid path of the file or directory you want to delete from the DGAS cache.

Step 6 If the target path is a directory and you want to delete it recursively, click the **Recursive?** box.

Step 7 Click **Submit**.

Saving a copy of the cache

Before performing this procedure, do the following:

- Follow the steps in “[Viewing and modifying cache statistics](#)” on page 123. Make a note of the total cache size.
- Ensure that there is enough free disk space on the machine where you plan to put the cache file.

Follow these steps to save a copy the DGAS cache to a local file.

Step 1 Log in to Avaki as a member of the Administrators group.

Step 2 Navigate to the View DGASes screen:

Home > Server management > View DGASes

View DGASes

You are viewing the DGASes in the local grid domain.

Server Name	Configure	Cache	Mappings	Policy	Destroy	Attributes	Security
dgas1	Configure	Cache	Mappings	Policy	Destroy	Attributes	Security

Step 3 Click the Cache link for the DGAS whose cache you want to configure. The system displays the Manage DGAS Cache screen:

Manage DGAS Cache

DGAS: dgas1

Manage Internal Cache Service

- [Clear credentials](#) Clear all cached user credentials
- [Delete from cache](#) Delete a grid directory or file from DGAS cache
- [Save cache](#) Save a copy of the cache for a DGAS
- [Sync file with proxy cache](#) Force a DGAS to fetch grid directories or files from proxy cache if the cached version is newer than the version on the DGAS; the DGAS will copy the directory or file if it is not cached
- [View/modify cache statistics](#) View and reset all statistics counters for the DGAS cache

Manage External Cache Service

- [Set external cache service](#) Associate an external proxy cache service with a DGAS

Step 4 Click **Save cache**. The Save Cache screen appears.

- Step 5** In the Local destination field, enter the path to a local file in which to save the current DGAS cache.
- Step 6** Click **Submit** to save the cache to the specified file.

Syncing the cache

Follow these steps to force the DGAS to fetch specified files or directories into its internal cache if it does not have up-to-date copies. The data may be fetched from the cache service associated with the DGAS, or, if the cache service does not have up-to-date copies, from the source file system.

- Step 1** Log in to Avaki as a member of the Administrators group.
- Step 2** Navigate to the View DGASes screen:

Home > Server management > View DGASes

View DGASes

You are viewing the DGASes in the local grid domain.

Server Name	Configure	Cache	Mappings	Policy	Destroy	Attributes	Security
dgas1	Configure	Cache	Mappings	Policy	Destroy	Attributes	Security

- Step 3** Click the Cache link for the DGAS whose cache you want to configure. The system displays the Manage DGAS Cache screen:

Manage DGAS Cache

DGAS: dgas1

Manage Internal Cache Service

Clear credentials	Clear all cached user credentials
Delete from cache	Delete a grid directory or file from DGAS cache
Save cache	Save a copy of the cache for a DGAS
Sync file with proxy cache	Force a DGAS to fetch grid directories or files from proxy cache if the cached version is newer than the version on the DGAS; the DGAS will copy the directory or file if it is not cached
View/modify cache statistics	View and reset all statistics counters for the DGAS cache

Manage External Cache Service

Set external cache service	Associate an external proxy cache service with a DGAS
--	---

- Step 4** Click **Sync file with cache**. The system displays the Sync Cache screen:

Sync Cache

A data grid access server can fetch a grid directory or file from cache if the cached version is newer than the version on the DGAS. The DGAS will copy the directory or file if it is not cached.

To force a DGAS to fetch the newest version or copy an uncached directory or file, specify the grid path of the directory or file to be fetched into cache. If the contents of a directory should be fetched and cached recursively, place a check mark next to **Recursive**.

DGAS name: dgas1

Target path:

Recursive?

Copyright © 2002-2003 Avaki Corp.

- Step 5** In the Target path field, enter the grid path to the file or directory you want the DGAS to fetch into its internal cache.
- Step 6** If the target path is a directory and you want to fetch all its contents, click the Recursive? box.
- Step 7** Click **Submit**.

Viewing and modifying cache statistics

Follow these steps to display or reset DGAS cache statistics. Cache statistics include cache size, free disk space, and hits and misses. (A cache hit occurs when a user calls for a file that's in the cache; a cache miss occurs when a user calls for a file that isn't in the cache.)

Step 1 Log in to Avaki as a member of the Administrators group.

Step 2 Navigate to the View DGASes screen:

Home > Server management > View DGASes

View DGASes

You are viewing the DGASes in the local grid domain.

Server Name	Configure	Cache	Mappings	Policy	Destroy	Attributes	Security
dgas1	Configure	Cache	Mappings	Policy	Destroy	Attributes	Security

Step 3 Click the Cache link for the DGAS whose cache you want to configure. The system displays the Manage DGAS Cache screen:

Manage DGAS Cache

DGAS: dgas1

Manage Internal Cache Service

- [Clear credentials](#) Clear all cached user credentials
- [Delete from cache](#) Delete a grid directory or file from DGAS cache
- [Save cache](#) Save a copy of the cache for a DGAS
- [Sync file with proxy cache](#) Force a DGAS to fetch grid directories or files from proxy cache if the cached version is newer than the version on the DGAS; the DGAS will copy the directory or file if it is not cached
- [View/modify cache statistics](#) View and reset all statistics counters for the DGAS cache

Manage External Cache Service

- [Set external cache service](#) Associate an external proxy cache service with a DGAS

Step 4 Click **View/modify cache statistics**. The system displays the View Cache Statistics screen.

Step 5 Click **Reset Statistics** to set the counters to zero, or click **Done** to return to the Manage DGAS Cache screen.

DGAS properties and cache size

Each DGAS maintains caches both in memory and on disk. The size and behavior of both memory and disk caches is controlled by several DGAS properties. This section describes the interactions of those properties.

You can display descriptions and current values of DGAS properties in both the web UI and the CLI:

- In the web UI, navigate to the View DGASes screen: Home > Server management > View data grid access servers, then click the Configure link for the DGAS whose properties you want to see.

- In the CLI, enter a command of this form:

```
avaki dgas --get-properties <dgas-name>
```

DGAS memory cache usage. Memory cache usage is controlled by several DGAS properties:

- Size of each fragment (this is the value of the cache-block-size property divided by the value of the cache-frags-per-block property)
- Maximum number of files cached (this is the value of the max-cached-files property)
- Number of fragments in memory per file (this is the value of mem-cache-stages)

Calculate the maximum size of the memory cache:

$$\frac{\text{cache-block-size}}{\text{cache-frags-per-block}} \times \text{max-cached-files} \times \text{mem-cache-stages}$$

If the maximum size of the memory cache is larger than 448MB, it may be necessary to adjust the Java max heap size for the DGAS. Contact Avaki customer support for assistance with this.

DGAS disk cache usage. Disk cache usage is also controlled by several DGAS properties:

- Cache block size (the value of the cache-block-size property)
- The number of cache blocks in the disk cache that are considered active (the value of the num-active-blocks property)
- The amount of space the disk cache will use for inactive blocks (the value of the inactive-blocks-high-water property, which is a value in bytes)

The maximum amount of space the disk cache can use is

$$(\text{cache-block-size} \times \text{num-active-blocks}) + \text{inactive-blocks-high-water}$$

It is important to set the `inactive-blocks-low-water` property to a value that's 70% to 90% of the value of `inactive-blocks-high-water`. If the DGAS reaches the maximum allowable disk cache size, it begins a process of removing the least recently used data from cache until the inactive cache is at the `inactive-blocks-low-water` mark. Setting this value incorrectly will result in inefficient DGAS cache management.

Creating and managing CIFS shares

This section explains how to set up and manage CIFS shares. A CIFS share makes data grid files and directories accessible to CIFS clients (usually Windows machines).

By default, a data grid access server (DGAS) that's configured to support CIFS shares exports two shares:

- A public share that's visible to all users and links to the grid's `/Shares` directory. This CIFS share is called `SHARES`.
- A hidden share called `AVAKIGRID$` that corresponds to the root of the grid data catalog (that is, the grid domain's entire file system). This share can be mounted explicitly by anyone who knows about it and has the appropriate permissions, but it does not appear in lists of available CIFS shares. Mount this share only on machines that need access to grid system data, such as the `/Metadata` grid directory, which houses schemas for Avaki database operations and data services.

You can also export specific grid directories by setting up CIFS shares for them, as described here.

Prerequisites

To create a CIFS share,

- You must set up a DGAS and configure it to support CIFS shares. See [“Setting up data grid access servers” on page 62](#) for details.
- You need execute permission on the DGAS that will serve the CIFS share and read permission on the grid directory that you are sharing out.

Creating a CIFS share









Follow these steps to create a CIFS share.

- Step 1** Log in to the grid domain as a user who has the permissions described under “Prerequisites,” above.
- Step 2** Navigate to the Browse Directories screen:
Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>		/	Attributes	Security	
<input type="checkbox"/>		GeneratedViews	Attributes	Security	Categories
<input type="checkbox"/>		Interconnects	Attributes	Security	Categories
<input type="checkbox"/>		Metadata	Attributes	Security	Categories
<input type="checkbox"/>		Shares	Attributes	Security	Categories
<input type="checkbox"/>		System	Attributes	Security	Categories
<input type="checkbox"/>		WSDLs	Attributes	Security	Categories
<input type="checkbox"/>		Categories	Attributes	Security	Categories

- Step 3** Click on directory names (starting with Shares, for example) to navigate to the grid directory that you want to export as a CIFS share.

Step 4 Click **Create CIFS Share**. The Create CIFS Share screen appears:

The screenshot shows a dialog box titled "Create CIFS Share". It contains the following fields and controls:

- Share name:** An empty text input field.
- Source path:** A text field containing the value "/Shares/Water Buffalo Lodge".
- Enabled?:** A checkbox that is currently checked.
- Maximum connections:** A text input field containing the value "100".
- Description:** A large, empty text area for entering a description.
- Buttons:** Two buttons at the bottom: "Continue" and "Cancel".

Step 5 Fill in the form:

- **Share name:** Enter a name for this CIFS share. (This name will be converted to all upper case—it's treated as a Windows file name and Windows is not case sensitive.)
- **Enabled?:** Leave the box checked to enable the share. Click to uncheck the box to disable the share. (You can enable the share later.)
- **Maximum connections:** The maximum number of users that will be permitted to access this share at one time. To allow unlimited connections, set maximum connections to -1 .
- **Description:** Enter a description for this CIFS share.

Step 6 Click **Continue**. The Select DGASes screen appears.

Step 7 Click in the Select column to choose a DGAS for this CIFS share.

Step 8 Click **Submit**. The system creates the CIFS share.

Setting system properties

Java system properties are set on a per-server or per-client basis. They reside in a system properties file on each Avaki server or Avaki Studio machine, or in the starting script for a client. System properties must be read when the server, Studio instance, or client starts, and changes take effect at the next restart.

In general, you should not need to change system property values. The exceptions are `java.rmi.server.hostname` and `com.avaki.cache.maxReaderThreads`. We recommend that you consult Sybase technical support before changing the values of any other system properties.

System properties on servers and clients

Each system property can be set on particular Avaki components—Avaki servers, Avaki Studio, command clients, or JDBC clients. The property descriptions provided in the [“System property reference” on page 131](#) tell you which components each system property can be set on.

Server properties are set in the same way on all Avaki servers—see [“Setting system properties for Avaki servers and Avaki Studio,”](#) below, for instructions.

Client properties. You can set client properties on:

- Command clients. For instructions, see [“Setting system properties for command clients” on page 129](#).
- Avaki Studio. See [“Setting system properties for Avaki servers and Avaki Studio,”](#) below.
- JDBC clients. See the *Sybase Avaki EII API Guide*.
- Any Avaki server. A server uses client properties when it makes a call, acting as a client to another Avaki server. All Avaki servers can act as clients (though it’s rare for proxy servers to do so). Follow the instructions in [“Setting system properties for Avaki servers and Avaki Studio,”](#) below.

Setting system properties for Avaki servers and Avaki Studio

To set system properties on an Avaki server or Avaki Studio, you edit the system properties file. The path to the system properties file varies:

```
<Avaki-install-dir>/jboss/server/grid-server/conf/system.properties
<Avaki-install-dir>/jboss/server/proxy-server/conf/system.properties
<Avaki-install-dir>/jboss/server/share-server/conf/system.properties
<Avaki-install-dir>/DGAS/system.properties
<Avaki-install-dir>/avaki_studio.properties
```

where

<Avaki-install-dir> is the local directory on each server machine in which Avaki software is installed.

Note There is a reference copy of the system properties file at <Avaki-install-dir>/docs/system.properties. This inactive version of the file contains descriptions and settings (commented out) for all system properties. When you set a property, you might find it convenient to copy the property's entry from the reference version of system.properties to the active system properties file.

Each Avaki component uses the default values for all system properties unless other values are set in the system properties file. Lines in the system properties file that begin with the # symbol (octothorpe) are comments; the system ignores them.

The syntax for setting properties in the system properties file is

```
<property-name> = <value>
```

For example, in a text editor, add this line to system properties:

```
com.avaki.retryDelay = 100
```

To make changes to system properties take effect, restart the server.

For details on individual system properties, see the [“System property reference” on page 131](#).

Setting system properties for command clients

To set system properties for an Avaki command client, you add each property to the SYSTEM_PROPS variable in the client's start-up script. Add a property in this format:

```
"-Dproperty-name=value"
```

Enclose each property/value pair in double quotes and separate each pair from the one following with a space. Examples for Windows and Unix scripts appear below.

For details on individual system properties, see the [“System property reference” on page 131](#).

In Windows. In a Windows installation, you set client properties by editing the file:

```
<Avaki-install-dir>\avaki.bat
```

For example, to add the DBOP Protocol SO Timeout property, edit the SYSTEM_PROPS variable in avaki.bat as follows:

```
set SYSTEM_PROPS="-Dcom.avaki.DBOProtocolSoTimeout=1200000"
"-Djava.library.path=%JAVAKI_HOME%\lib" "-Dcom.avaki.InstallDir=%JAVAKI_HOME%"
-Dcom.avaki.Version=%AVAKI_VERSION% -Dcom.avaki.InstallType=client
"-Djavax.net.ssl.trustStore=%JAVAKI_HOME%\resources\cacerts"
"-Dcom.avaki.AdgSessionDir=%ADG_SESSION_DIR%"
-Djava.protocol.handler.pkgs="|com.avaki.core.net"
```

In Unix (including Linux). In a Unix installation, you set client properties by editing the file

```
<Avaki-install-dir>/avaki
```

For example, to add the DBOP Protocol SO Timeout property, edit the SYSTEM_PROPS variable in the avaki script as follows:

```
SYSTEM_PROPS="-Dcom.avaki.DBOProtocolSoTimeout=1200000"
"-Djava.library.path=\"\$JAVAKI_HOME/lib\" -Dcom.avaki.InstallDir=\"\$JAVAKI_HOME\" "
-Djava.protocol.handler.pkgs=|com.avaki.core.net
-Dcom.avaki.Version=$AVAKI_VERSION -Dcom.avaki.InstallType=client
\"-Djavax.net.ssl.trustStore=$JAVAKI_HOME/resources/cacerts\"
-Dcom.avaki.AdgSessionDir=\"\$ADG_SESSION_DIR\""
```

System property reference

This section describes system properties you can set on Avaki servers, clients, and on Avaki Studio.

Setting a server's host name or IP address

By setting a Java system property called `java.rmi.server.hostname`, you can specify a DNS name or IP address for your Avaki GDC, grid server, DGAS, proxy server, or share server to advertise.

For instructions on setting `java.rmi.server.hostname`, see [“Setting a server's host name or IP address” on page 32](#).

Note You can find an Avaki server's name in its `connectinfo.txt` file. The path to this file varies by server type:

- <Avaki-install-dir>/jboss/server/grid-server/log/connectinfo.txt
- <Avaki-install-dir>/jboss/server/proxy-server/log/connectinfo.txt
- <Avaki-install-dir>/jboss/server/share-server/log/connectinfo.txt
- <Avaki-install-dir>/dgas_connectinfo.txt

(<Avaki-install-dir> is the local directory where Avaki software is installed.)

Configuring read threads for cache services

Proxy cache services provide the results of data services and database operations to client applications (via the web services API or JDBC) and to Avaki views. To get the most from your execution services, data services and database operations, you can configure a Java system property that controls the number of client applications and views that can simultaneously read data service and database operation results from a cache service.

The system property is `com.avaki.cache.maxReaderThreads`. By default, this property allows unlimited reader threads. If you wish to limit the number of reader threads, read this section for instructions on calculating an appropriate setting for the property.

`Com.avaki.cache.maxReaderThreads` specifies the maximum number of clients that can concurrently read cached data service results or database operation results from the cache service on this grid server. Read requests in excess of this number are queued; the requestors in the queue receive results when another requestor finishes reading. Each requestor is assigned its own thread. Set this value to a number of

threads (requestors) that will not overtax the grid server machine. (The thread capacity of a machine is determined by a combination of factors including processing power, memory, and limits imposed by the operating system.)

The default value of `com.avaki.cache.maxReaderThreads` is `-1` (unlimited reader threads).

If you change the value of `com.avaki.cache.maxReaderThreads`, the value you choose should include enough threads for the views and client applications that will call on this cache service. If the value is too low, performance will be poor: users and applications will have to wait for the results of database operations and data services. If the value is too high—that is, if too many views and client applications are using this cache service—performance will also be poor: the machine will not be able to keep up with the demand.

To calculate a value, add the maximum concurrent data services settings (found on the View/Edit Execution Service screen) for each execution service on this grid server, including pooled execution services. For example, if this grid server hosts a pool of three execution services that each have a maximum concurrent data services value of 10—meaning that you expect up to 30 client applications to simultaneously request uncached data service results—you should set `com.avaki.cache.maxReaderThreads` no lower than 30.

For best results, include additional threads for reading cached data service results and both cached and uncached database operation results. For example, suppose that you expect up to 10 concurrent calls for cached data service results (from client applications and other data services). Suppose further that this cache service must accommodate requests for database operation results from 30 views and 10 data services. The views and data services are on staggered schedules and rarely run simultaneously, so five threads each for uncached and cached database operation results will be ample.

Our calculation is

Threads for reading uncached data service results (requires running data services in execution services)	30
Threads for reading cached data service results	10
Threads for reading uncached dbop results	5
Threads for reading cached dbop results	5
Total	50

Setting message timeout properties

You can set timeout properties for the messages that Avaki servers use to communicate with one another over your network. The default values of these properties work well for most servers, and Avaki does not recommend changing them. However, if your network performance is unusually good or bad, you might be able to improve grid performance by changing timeout values.

- If your network is very reliable and fast, consider lowering your timeout values. This will allow your grid to detect failures faster and improve performance.
- If your grid works over a wide-area network or suffers from connections that are unreliable or simply slow, consider raising your timeout values. This will decrease the chances that an apparent problem in the grid is actually a problem in the network.

Caution We recommend that you consult Sybase technical support before changing message timeout property values. Changing these values without a good understanding of data grid architecture, the properties themselves, and the configuration of your network can cripple or disable your data grid.

Note When a message timeout property is set to 0, it never times out. Do not set message timeout properties to negative values.

These are the message timeout properties.

Property	Description	Default Value
com.avaki.retryTimeout	The time in milliseconds to retry calls that fail as a result of errors before giving up. Set on all servers and clients.	10000 (10 seconds)
com.avaki.retryDelay	The time in milliseconds to pause between retries. Set on all servers and clients.	250 (0.25 seconds)
com.avaki.proxy.retryTimeout	The period in milliseconds during which the proxy client keeps trying to send calls to the proxy server. Set on servers that act as proxy clients.	10000 (10 seconds)
com.avaki.proxy.retryDelay	The period in milliseconds during which the proxy client pauses between retries. Set on servers that act as proxy clients.	250 (0.25 seconds)

Property	Description	Default Value
com.avaki.DBOProtocolSo- Timeout	The period in milliseconds after which, if there is no response, this client will terminate a call to the server for the result of a database operation. Set on all servers and clients.	600000 (10 minutes)
com.avaki.proxyIOProtocol- SoTimeout	The period in milliseconds after which a blocking proxy SSL I/O socket call (to a proxy server) times out. This timeout applies only to RPCs over SSL-enabled proxies. Clear-enabled proxies have no socket timeout property. Set on all servers except proxy servers.	30000 (30 seconds)
com.avaki.shareIOProtocol- SoTimeout	The period in milliseconds after which a blocking share I/O socket call (reading the contents of a shared file) times out. Set on share servers and their clients.	30000 (30 seconds)
com.avaki.rpcTimeout	The period in milliseconds after which a blocking RMI RPC socket call times out. This property affects all communication between Avaki objects that reside on different Avaki servers. If your long-running RPCs (such as blocking share creation) are timing out because they're running longer than 2 hours (or the present value of this property), increase this value. To reduce the time it takes an RPC to time out and fail, set this property to a lower value. Set on all servers and clients. Note that this property interacts with com.avaki.mux.channelSoTimeout , below.	7200000 (2 hours)
com.avaki.rmiRegistrySo- Timeout	The period in milliseconds after which a blocking RMI registry lookup socket call times out, throwing a java.io.InterruptedIOException. RMI registry lookups obtain initial connection endpoint information for communication between Avaki servers. These operations are short-lived and should have a much shorter timeout than the general RPC timeout, com.avaki.rpcTimeout (above). Set on all servers and clients.	30000 (30 seconds)
com.avaki.mux.connect- Timeout	The period in milliseconds that a multiplexing client socket will wait for a new real connection to a server. The connection timeout is for the multiplexing socket's underlying real socket. This property does not affect existing connections. Consider increasing the value of this property if you see timeout exceptions on the client even though the server is running. The valid values are 0 (never times out) to MAXINT. For more information on multiplexing properties, see " Setting multiplexing socket properties " on page 142. Set on all servers and clients.	2000 (2 seconds) for DGAS; 3000 (3 seconds) for grid servers, share servers, CLI and Studio

Property	Description	Default Value
<code>com.avaki.mux.channelSocketTimeout</code>	<p>The period in milliseconds after which a blocking multiplexing channel socket call times out. The value of this property:</p> <ul style="list-style-type: none"> • Is the connect timeout for virtual sockets. • Affects the write timeout for virtual sockets. Write operations on virtual sockets fail when the virtual socket timeout, which defaults to the value of <code>com.avaki.rpcTimeout</code>, is reached. However, the virtual socket's timeout is rounded up to the nearest multiple of the channel socket's timeout (that is, to the nearest multiple of the value of this property). For example, every setting of <code>com.avaki.rpcTimeout</code> from 30001 to 60000 results in a 60-second write timeout, assuming this property is set to its default value. <p>Set on all servers and clients.</p>	30000 (30 seconds)

Restart the Avaki component (server, client, or Studio) to make changes to property values take effect.

Setting the location of the temp directory

`java.io.tmpdir`

Default value: `<Avaki-install-dir>/jboss/server/grid-server/tmp`

Set on: grid servers

Use `java.io.tmpdir` to set the path to a directory that this grid server can use to write temporary files. The grid server writes temporary files when it executes data services that perform sorting.

Setting cache service properties

See also “[Configuring read threads for cache services](#)” on page 131.

`com.avaki.cache.cacheDir`

Default value: `<Avaki-install-dir>/<host-name>-cache`

Set on: grid servers

Specifies the local directory used by the cache service associated with this server.

com.avaki.cache.writeInvalidationQueueSize

Default value: 256

Set on: clients

Sets the maximum number of items allowed in the write invalidation queue for the cache service associated with this client. Invalidation requests for file objects are placed in the queue when this client writes to a cached file.

Invalidation can happen either (a) immediately after the write (so that all other cache clients will get the new data) or (b) as part of the read request the next time this client attempts to read the updated file from the cache service (so that at least this particular client has a consistent view). The invalidation policy of (a) can bog down performance with extra RPCs, while that for (b) requires memory to remember which files need to be invalidated the next time they are read.

The invalidation queue guarantees write-then-read consistency for the server. A value of 0 means that cached items will be invalidated immediately after they have been written to, which ensures consistency for all clients of the associated cache service.

If the value of this property is not 0, the client maintains a list of items to invalidate the next time they are read from. If the size of this list of files to invalidate grows past the value of this property, a separate RPC is made to the associated cache service to invalidate all the items in the list. This prevents the list from growing without bounds if the client writes to a lot of files and never reads from them.

com.avaki.maxActiveCachables

Default value: 256

Set on: grid servers

Specifies the maximum number of persistent representations of cached objects that can be stored in the state cache in the cache service on this server.

com.avaki.result.gcInterval

Default value: 30

Set on: grid servers

Sets the frequency, in minutes, at which the cache service on this grid server cleans up files containing expired data from the results of database operations and data services.

com.avaki.vaultStateCacheSize

Default value: 5000

Set on: grid servers

Specifies the maximum number of persistent representations of grid objects that can be stored in the state cache in the vault on this server.

Setting cache sizes for data service plug-ins

com.avaki.dataservice.pluginCacheSize

Default value: 24

Set on: grid servers

Controls the number of data service plug-ins written in Java or JavaScript that a grid server can store in memory. (Plug-ins for data services created in Avaki Studio are written in JavaScript.) When this maximum value is reached, the least recently used plug-in is discarded to make room for the new plug-in.

com.avaki.dataservice.styleSheetCacheSize

Default value: 64

Set on: grid servers

Controls the number of XSLT data service plug-ins that a grid server can store in memory. When this maximum value is reached, the least recently used stylesheet is discarded to make room for the new stylesheet.

com.avaki.dataservice.styleSheetCachePoolSize

Default value: 5

Set on: grid servers

Controls the number of instances of a each XSLT data service plug-in that a grid server can store in memory. Multiple copies improve response time for concurrent accesses of the stylesheet.

Setting share server properties

See also “[com.avaki.shareIOProtocolSoTimeout](#)” on page 134.

com.avaki.shareServerCircularLinkChecking

Default value: true

Set on: share servers running on Unix

Enables and disables detection of circular links in Unix directory hierarchies that are shared into a data grid. If the share server finds an infinite loop in a shared directory, it logs a warning and prevents the Avaki share from recursing down that path when it refreshes. To disable circular link checking, set to “false”. This property has no effect on Windows-based share servers.

com.avaki.shareServerThreadPoolMaxSize

Default value: -1

Set on: share servers

Controls the maximum number of threads available to handle concurrent requests to this share server. A value of -1 means that thread pool size is unlimited.

com.avaki.shareServerThreadPoolMinSize

Default value: 5

Set on: share servers

Controls the minimum number of threads available to handle concurrent requests to this share server.

com.avaki.shareReadBufferSize

Default value: 1048576

Set on: share servers

Controls the size in bytes of the read buffers used by this share server. (The read buffer reduces the frequency of memory allocations and garbage collection.) Read buffers are also controlled by [com.avaki.shareReadBufPoolSize](#), below.

com.avaki.shareReadBufPoolSize

Default value: 5

Set on: share servers

Controls the number of read buffers that this share server can keep in memory. If there are not enough read buffers to service every thread, other memory is allocated on the fly. Multiply this value by the value of [com.avaki.shareReadBufferSize](#), above, to

determine how much memory is set aside for read buffers. This value should be no greater than the maximum number of share server threads (that is, the value of [com.avaki.shareServerThreadPoolMaxSize](#), above).

Setting the chunk size for query engine sort operations

com.avaki.queryEngine.sortChunkSize

Default value: 10,000

Set on: grid servers

To sort arbitrarily large result sets, the query engine breaks sort inputs into chunks and uses the local disk to hold intermediate results. (These intermediate results are cleaned up when the operation completes.) This property lets you set the chunk size in rows.

There is a trade-off between the amount of memory that a grid server uses for its computation and the amount of I/O that it must perform to sort or join large result sets. The query engine can break the inputs into a large number of small chunks or into a smaller number of larger chunks. A large number of small chunks requires less memory at any given time, but will result in more disk I/O. A small number of large chunks will result in a larger peak memory usage, but less disk I/O overall.

Setting encryption for grid objects

com.avaki.content.encryptionLevel

Default value: clear

Set on: grid servers

Specifies whether database operations and data services created on this grid server use SSL encryption when they send their results over the network. Accepted values: clear, encrypted.

Setting HTTP ports

The HTTP and HTTPS ports must be changed in `bindings.xml`, the Avaki server's configuration file—changing the system properties is not necessary.

com.avaki.HttpPort

Default value: varies by server type

Set on: all servers except share servers and DGAS

The HTTP port used by this Avaki server. For the default HTTP port for each server type, see [“Planning use of network ports” on page 8](#).

com.avaki.HttpsPort

Default value: varies by server type

Set on: all servers except share servers and DGAS

The HTTPS port used by this Avaki server. For the default HTTPS port for each server type, see [“Planning use of network ports” on page 8](#).

Setting HTTP keepalive parameters

com.avaki.proxy.httpKeepAliveParams

Default value: none (pass no parameters)

Set on: proxy servers

Passes two HTTP keepalive parameters that are accepted by Sun HTTP clients. The parameters are

- `timeout`: The amount of time in seconds to maintain idling connections.
- `max`: The maximum allowed number of idling connections.

To set the two parameters, set this property according the format in the following example:

```
com.avaki.proxy.httpKeepAliveParams=timeout=15,max=10
```


Setting client properties

See “[System properties on servers and clients](#)” on page 128 for a discussion of what “client” means in this context.

java.protocol.handler.pkgs

Default value: |com.avaki.core.net

Set on: clients

Invokes the Avaki URL stream handler to handle the avaki protocol when it is specified in a URL. Do not change the value of this property.

com.avaki.badPortCacheSize

Default value: 100

Set on: clients

Controls the number of items stored in the LRU cache that remembers bad port locations. Each entry in the BadPortCache is a URL string (for example, `jnp://192.168.68.42:3099//Avaki/ConfigurationServiceContainer`). The cache remembers the location information: host and port, protocol, and the object container type. If a port is in the BadPortCache, the client tries other ports first, but if necessary it will try all ports, including any in the cache.

com.avaki.badPortExpiration

Default value: 5000 (5 minutes)

Set on: clients

Controls the time-to-live value in milliseconds of the entries in the BadPortCache—that is, how long this client keeps information about bad ports it has encountered.

Setting the remote coherence window for configurations

com.avaki.remoteconfig.coherenceWindow

Default value: 1800 (30 minutes)

Set on: proxy servers and DGASes

Controls the time in seconds during which this server assumes that cached remote configuration objects are fresh. During this period, this server does not go to the source to obtain new configurations.

Setting the XML indent size

com.avaki.generatedXMLIndentSize

Default value: 2

Set on: grid servers and clients

Controls the size in spaces of indents used in XML documents generated in Avaki

Setting multiplexing socket properties

Avaki is configured by default to use the `com.avaki.core.util.mux.MUXSSLFactory` / `com.avaki.core.util.mux.MUXSSLSocketFactory` `RMISocketFactories` for communication. These socket factories create virtual sockets that are multiplexed over a single TCP channel socket in order to reduce the number of open sockets/file descriptors and to improve communication performance by reducing socket setup/teardown overhead.

In addition to the properties listed here, see these multiplexing socket timeout properties:

- [com.avaki.mux.connectTimeout](#) on page 134
- [com.avaki.mux.channelSoTimeout](#) on page 135

com.avaki.mux.maxParallelChannels

Default value: 1

Set on: clients

Sets the maximum number of parallel channels that a client (which may be an Avaki server) can open to the same endpoint. Parallel channels can improve throughput, which can be useful for Avaki servers (particularly DGASes and grid servers with active cache services) that make many concurrent RPCs to nonlocal destinations. Suggested range: 1 to 4 (Values above 4 have not been shown to improve throughput.)

com.avaki.mux.maxWriteChunk

Default value: 262144 (256 kB)

Set on: clients

Sets the maximum number of contiguous bytes the multiplexing socket's output stream is allowed to write over the channel's socket. (Byte arrays larger than specified here are broken into multiple writes.) Reduce this value if some virtual connections seem to be starving others by writing large amounts of data over high-latency channels.

com.avaki.mux.sendBufferSize

Default value: 65536

Set on: all servers

Sets the send buffer size, in bytes, for the multiplexing socket's TCP socket.

Setting Kerberos properties

You must set Kerberos system properties if you configure an LDAP authentication service to use Kerberos. For instructions, see [“Configuring Kerberos access” on page 152](#).

java.security.krb5.conf

Default value: <JRE-install-dir>/lib/security. If the file is not there, the system looks for it in other locations that vary by Kerberos implementation.

Set on: the machine that hosts the LDAP authentication service (usually the GDC)

The path to krb5.conf, the Kerberos configuration file.

java.security.krb5.kdc

Default value: none

Set on: the machine that hosts the LDAP authentication service (usually the GDC)

The IP address or DNS name of the machine that hosts the Kerberos key distribution center. If you set this property, you must also set java.security.krb5.realm. Set these two properties only if you don't have a Kerberos configuration file.

java.security.krb5.realm

Default value: none

Set on: the machine that hosts the LDAP authentication service (usually the GDC)

The name of the Kerberos default realm. If you set this property, you must also set java.security.krb5.kdc. Set these two properties only if you don't have a Kerberos configuration file.

Setting the cache size for the virtual database

com.avaki.VirtualDbTableCacheSize

Default value: 256

Set on: grid servers

Controls the number of table representations that a grid server's virtual database can store in memory. When the contents of the table cache reach this maximum value, the least recently used table representation is discarded to make room for the new one.

Setting properties for the remote object stub cache

When a client makes a remote method call into a grid server, the client must obtain a stub or handle to reach the targeted object. A stub that's targeted to a remote object in a grid server can be reused by any other invocations from the client to the same object. To reduce the overhead of obtaining stubs, each client (including servers) maintains a pool of stubs for each targeted Avaki server; the remote object stub cache stores the most recently used pools.

com.avaki.lasInvoker.cacheSize

Default value: 20

Set on: clients

Specifies the maximum number of remote object stub pools that the remote object stub cache will save.

com.avaki.lasInvoker.poolSize

Default value: 5

Set on: clients

Specifies the maximum number of stubs that can be stored in each remote object stub pool on a client.

Setting properties for the schedule exclusion cache

Each grid server's scheduler maintains a schedule exclusion cache. The schedule exclusion cache is only used when schedule exclusion information cannot be obtained directly from the exclusion's associated grid object stored on the GDC. The cache allows non-GDC grid servers' schedulers to run smoothly even if the GDC is down or there is a communication outage between the grid server and the GDC.

com.avaki.scheduleExclusionCacheSize

Default value: 10

Set on: grid servers

Specifies the number of elements that can be saved in the schedule exclusion cache. A setting of 0 (zero) disables schedule exclusion caching.

com.avaki.scheduleExclusionCacheExpiration

Default value: 604800 (1 week)

Set on: grid servers

Specifies the number of seconds after a schedule exclusion entry is added to the cache that its entry expires.

Setting the operations status expiration interval

com.avaki.jobStatusExpiration

Default value: 15

Set on: grid servers

Specifies the number of minutes for which the operations status service continues to report on operations (including data services, database operations, share rehashes, and view generators) that have finished running.

Setting the page size for LDAP results

com.avaki.ldap.resultPageSize

Default value: 1000

Set on: the machine that hosts the LDAP authentication service (usually the GDC)

Specifies the number of entries per “page” the LDAP server will return when you import users or groups. If the result doesn’t fit in one “page”—that is, if the number of users or groups exceeds the number of entries specified here—the LDAP authentication service retrieves additional pages until all entries are read. Valid values are 1 to MAXINT; do not set this property to values below 1.

Setting the TDS port

com.sybase.avaki.tdsPort

Default value: 15000

Set on: GDC only

Specifies the GDC port from which TDS packets are consumed. The TDS protocol enables communication to Avaki from the jConnect JDBC driver, the ASE ODBC Organic driver, and any other ct-lib based client application.

Managing authentication services, users, and groups

This chapter covers the following topics:

- [“Managing authentication services,”](#) below
Includes integrating LDAP and NIS directory services into the grid, importing user accounts and groups from authentication services, and deleting authentication services.
- [“Managing users” on page 167](#)
Includes creating and deleting grid user accounts, refreshing imported accounts, setting up home directories, and changing passwords.
- [“Managing groups” on page 191](#)
Includes creating and deleting grid groups, refreshing imported groups, adding users to groups and removing users from groups.

For introductory information about authentication in Avaki domains and descriptions of the default groups in a domain, see the *Sybase Avaki EII Overture*.

Which administration group? There are two default groups for users who perform administrative functions in the data grid: Administrators and UserAdministrators. The procedures in this chapter tell you which group to use for each task.

Before tackling most of the tasks described in this chapter, create or import user accounts for grid administrators and assign them to Administrators or UserAdministrators as described in [“Setting up administrative accounts in Avaki” on page 45](#).

Managing authentication services

This section covers the following topics:

- [“Integrating LDAP authentication services,”](#) below
- [“Configuring Kerberos access”](#) on page 152
- [“Importing user accounts from an LDAP authentication service”](#) on page 157
- [“Importing groups from an LDAP authentication service”](#) on page 159
- [“Integrating NIS authentication services”](#) on page 162
- [“Importing user accounts from an NIS authentication service”](#) on page 163
- [“Importing groups from an NIS authentication service”](#) on page 165
- [“Viewing existing authentication services”](#) on page 166
- [“Deleting authentication services”](#) on page 166

Integrating LDAP authentication services

Services based on based on the Lightweight Directory Access Protocol (LDAP) include Microsoft Active Directory or the Sun ONE Directory Server (formerly iPlanet).

To integrate an LDAP-based directory service into your grid domain and create a corresponding authentication service in the grid, follow these steps.

Note We recommend that this procedure be performed by or in close consultation with the LDAP administrator.

- Step 1** Log in to Avaki as a member of the Administrators group.
- Step 2** Navigate to the Integrate LDAP Authentication Service (step 1) screen:

Home > User management > Integrate LDAP authentication service

Integrate LDAP Authentication Service (step 1)

For "Service name," specify a name for the LDAP authentication service. This service name, which can be any combination of letters or numbers, will appear in the list of authentication services that users can log in to.

Service name:

LDAP machine (DNS name or IP address):

Refresh users on login:

Authentication method:

Authentication server distinguished name:
(example: CN=Joe Smith, OU=Burlington, DC=Avaki, DC=Local)

Authentication server password:

User account attribute:

Group name attribute:

(optional) UID attribute:

(optional) Enabled query:

Step 3 Fill in the form:

- Service name: Devise a name for the new service and enter it here.
- LDAP machine (DNS name or IP address): Specify the name or IP address of the host on which the LDAP server resides.

Note Avaki uses the default LDAP host port, 389, to communicate with the LDAP server. If your LDAP host uses another port, you cannot perform the Avaki-LDAP integration using Avaki's web UI—use the **avaki ldap --integrate** command instead. The command lets you specify an alternate LDAP host port. See the *Sybase Avaki EII Command Reference* for details on using **avaki ldap --integrate**.

- Refresh users on login: If you check this box, each user account imported through this authentication service will be reimported every time the user logs in. Checking this box is a good way to ensure that changes to an imported user account, such as group membership changes, are available in the grid. If you do not check this box, we recommend that you do one of the following:
 - Configure a schedule to refresh the authentication service.

- Periodically reimport user accounts manually—see [“Keeping authentication services up to date” on page 185](#) for more information.
- Authentication method: From the pull-down menu, select the authentication method used by your LDAP server: simple, DIGEST-MD5, CRAM-MD5, EXTERNAL, or GSSAPI. If you select GSSAPI, you must perform additional configuration to enable Avaki to communicate with Kerberos—see [“Configuring Kerberos access” on page 152](#) for instructions.
- Authentication server distinguished name:
 - If you selected the GSSAPI authentication method, enter the Kerberos user ID for a user who has authority to search for and read all user and group entries in the LDAP directory. This user’s credentials will be used for all requests to the LDAP server from this authentication service. If you don’t know what to enter here, consult your LDAP administrator.
 - If you selected an authentication method other than GSSAPI, enter the distinguished name (DN) for a user who has authority to search for and read all user and group entries in the LDAP directory. This user’s credentials will be used for all requests to the LDAP server from this authentication service. For an Active Directory installation, the string you enter might include values for CN, OU, DC—for example, CN=Joe Smith, OU=Burlington, DC=Avaki, DC=com. For a Sun ONE (iPlanet) installation, the string might include values for UID, OU and O. If you don’t know what to enter here, consult your LDAP administrator.
- Authentication server password: Enter the password for the user specified in the authentication server distinguished name field, above.
- User account attribute: Enter the name of the LDAP attribute that contains the user ID for users from this authentication service. (This is where to find the user ID that each LDAP user will use to log in to this Avaki grid domain.) For example, you can enter sAMAccountName for an out-of-the-box Active Directory installation, or uid for an out-of-the-box Sun ONE (iPlanet) installation.
- Group name attribute: Enter the name of the LDAP attribute that contains the group name that users from this authentication service will use to log in to the Avaki grid domain. For example, you can enter “cn” for out-of-the-box Active Directory or Sun ONE (iPlanet) installations.
- UID attribute: Optional. Enter the name of the LDAP attribute that contains the Unix UID corresponding to each user. If users in this authentication service will access Unix files through an Avaki data grid access server (DGAS), you must provide mappings between user accounts known to the data grid and user accounts on the Unix

systems that host the files. If you don't provide this information through the LDAP directory, you can define it on a user-by-user basis later.

- **Enabled query:** Optional. Enter an LDAP query that finds out whether users in this directory are enabled. If you enter a query, the Avaki web UI and CLI display the most recent result when you view user accounts. If you don't enter a query, all user accounts in this authentication service will be reported "enabled." The query is run when the user account is imported or refreshed.

Step 4 Click **Continue** to save your entries; the Integrate LDAP Authentication Service (step 2) screen appears.

Integrate LDAP Authentication Service (step 2)

Enter user search base parameters (blank lines will be ignored):
 User Search Base 1:

Add User Search Base

Enter group search base parameters (blank lines will be ignored):
 Group Search Base 1:

Add Group Search Base

Submit **Cancel**

Step 5 Fill in the form:

- **User search base 1:** The top level of the first LDAP directory context within which the system will search for users. For example, you might enter OU=Burlington, DC=Avaki, DC=com. The system searches on the LDAP name components to find users that you can import. If you don't know what to enter here, consult your LDAP administrator.

To add more user search bases, click **Add User Search Base**. You can add as many as you need. User search base 2 is the top level of the second LDAP directory context within which the system will search for users; User search base 3 is the top level of the third, and so on.

- **Group search base 1:** The top level of the first LDAP directory context within which the system will search for groups. For example, you might enter OU=Groups, DC=Avaki, DC=com. The system searches on the LDAP distinguished name components to find groups that you can import. If you don't know what to enter here, consult your LDAP administrator.

To add more group search bases, click **Add Group Search Base**. You can add as many as you need. Group search base 2 is the top level of the second LDAP directory context within which the system will search for groups; Group search base 3 is the top level of the third, and so on.

- Step 6** Click **Submit** to integrate the authentication service. The system displays a summary of the information you entered.

For instructions on importing users and groups from a newly integrated authentication service, see [“Importing user accounts from an LDAP authentication service” on page 157](#) and [“Importing groups from an LDAP authentication service” on page 159](#).

Configuring Kerberos access

If you have configured an LDAP authentication service to use the GSSAPI authentication method, you must complete the following tasks so that Avaki software can communicate with Kerberos:

- Provide Kerberos information to the data grid. For details, see [“Providing Kerberos information,”](#) below.
- Provide a mapping from the LDAP distinguished names that the Avaki authentication service is using to appropriate Kerberos IDs. For details, see [“Mapping the LDAP DN to a Kerberos user ID” on page 154](#).

Providing Kerberos information

In order to perform authentication via Kerberos, Avaki software needs to know the Kerberos installation’s default realm and KDC (key distribution center). You can provide this information in either of the following ways:

- Make the Kerberos configuration file, `krb5.conf`, available to Avaki. Use this method if your Kerberos implementation has a configuration file. It’s described in [“Providing the Kerberos configuration file,”](#) below.
- If your Kerberos implementation does not include a `krb5.conf` file, you can make the default realm and KDC available by setting system properties. (Microsoft’s Kerberos for Windows 2000 is an example of an implementation that does not have a `krb5.conf` file by default.) For instructions on setting the system properties, see [“Setting Kerberos system properties” on page 153](#).

Providing the Kerberos configuration file. To make the Kerberos configuration file available to Avaki, follow these steps:

- Step 1** Put a copy of the Kerberos configuration file on the machine that hosts the LDAP authentication service (usually the GDC machine).

Note The default directory for the `krb5.conf` file is `<JRE-install-dir>/lib/security`. If you put the file in this location, you can skip the rest of this procedure.

- Step 2** If the `krb5.conf` file is not in its default location, use a text editor to open the `system.properties` file on the host machine of the LDAP authentication service. The `system.properties` file is in the `<avaki-install-dir>/jboss/server/grid-server/conf` directory.
- Step 3** In the `system.properties` file, set the `java.security.krb5.conf` property to the location of the Kerberos configuration file. For example:

```
java.security.krb5.conf = /etc/krb5.conf
```

- Step 4** To make the property change take effect, restart the grid server.

Setting Kerberos system properties. To make the Kerberos default realm and KDC available to Avaki via system properties, you must set the following properties:

- `java.security.krb5.realm`
- `java.security.krb5.kdc`

Follow these steps to set the properties:

- Step 1** In a text editor, open the `system.properties` file on the host machine of the LDAP authentication service. The `system.properties` file is in `<avaki-install-dir>/jboss/server/grid-server/conf`
- Step 2** In the `system.properties` file, set the `java.security.krb5.realm` property to the name of your Kerberos realm. For example:
- ```
java.security.krb5.realm = SLATE-QUARRY.COM
```
- Step 3** Set the `java.security.krb5.kdc` property to the IP address or DNS name of the machine that hosts the Kerberos key distribution center. For example:
- ```
java.security.krb5.kdc = 172.16.15.203
```
- Step 4** To make the property change take effect, restart the grid server.

Mapping the LDAP DN to a Kerberos user ID

To perform authentication via Kerberos, Avaki software needs to know the Kerberos ID to use for each user making requests to this LDAP authentication service.

There is a default mapping. When you integrated this authentication service into the data grid, you entered a user account attribute (see [page 150](#)) specifying the field that contains the user ID associated with each LDAP account.

By default, Avaki assumes that the user ID associated with each authentication service distinguished name (DN) is the same as the Kerberos ID. If this assumption is correct, you don't need to perform the procedure in this section. This default works with Active Directory's Kerberos system.

Otherwise, to enable Avaki to use Kerberos, you must add two attributes to the LDAP authentication service. The attributes establish a mapping from LDAP distinguished names to appropriate Kerberos IDs.

Follow these steps to add the new attributes.

Step 1 Navigate to the View Authentication Services screen:

Home > User management > Manage existing authentication services

View Authentication Services

The authentication services below have been integrated into your grid. To view additional information about a service, edit a service's properties, or import users and groups from a service, click on **View/Edit**.

Authentication Service Name	Type	View/Edit Service	Schedule	Delete Service	Attributes	Security
DefaultAuthService	Grid	View/Edit			Attributes	Security
testldap	LDAP	View/Edit	Schedule	Delete	Attributes	Security

Step 2 On the View Authentication Services screen, click the **Attributes** link to the right of the LDAP authentication service's name.

- Step 3** When the View Attributes screen appears, scroll to the User-defined Attributes area at the bottom.

User-defined Attributes

A user-defined attribute may be one of the following types:

Type	Description	Format
String	Text	Any characters
Integer	Any whole quantity	Any integer
Float	A numeric value that can be fractional or very large	Any numeric value
Date	Year, month, and date	yyyy-mm-dd
Time	Hour, minute and second that an event occurs	hh:mm:ss
Timestamp	A precise time and date	yyyy-mm-dd hh:mm:ss.ffffff

This item currently has no user attributes.

To add a new attribute, specify a name, type, and value in the fields below.

Name: Type: String Value:

- Step 4** To add the first attribute, enter the following in the Name field near the bottom of the User-defined Attributes area:

```
ldap/kerberos/regexp
```

- Step 5** In the Value field, enter a Perl5.003-style regular expression matching the entire LDAP DN. For example:

```
CN= ([^, ]+), \s*OU= ([^, ]+), \s*DC= ([^, ]+), \s*DC= (.+)
```

Avaki uses the Jakarta-ORO regular expression library for Perl5-style regular expressions. For information on building expressions for your regexp and idMap attributes, see the Jakarta-ORO Javadoc at

<http://archive.apache.org/dist/jakarta/oro/jakarta-oro-2.0.7.zip>

More information can be found in the **perlre** man page that accompanies Perl5 distributions and in *Programming Perl*, 2nd Edition, from O'Reilly & Associates. Note, however, that in Avaki attributes you should not use double (escaped) backslashes as recommended in the Javadoc (for example, use `\U`, not `\\U`).

Step 6 Click **Submit**. The new attribute is added to the User-defined Attributes table.

User-defined Attributes				
A user-defined attribute may be one of the following types:				
Type	Description	Format		
String	Text	Any characters		
Integer	Any whole quantity	Any integer		
Float	A numeric value that can be fractional or very large	Any numeric value		
Date	Year, month, and date	yyyy-mm-dd		
Time	Hour, minute and second that an event occurs	hh:mm:ss		
Timestamp	A precise time and date	yyyy-mm-dd hh:mm:ss.ffffff		
To modify an attribute's value, enter the new value and make sure the Update checkbox is selected. To delete an existing attribute, check the checkbox in its Delete column.				
Name	Type	Value	Update	Delete
ldap/kerberos/regexp	string	CN=([^\,]+),\s*OU=([^\,]+),\s	<input type="checkbox"/>	<input type="checkbox"/>
To add a new attribute, specify a name, type, and value in the fields below.				
Name:	<input type="text"/>	Type:	<input type="text" value="String"/>	Value: <input type="text"/>

Step 7 To add the second attribute, enter the following in the Name field:

ldap/kerberos/idMap

Step 8 In the Value field, enter an expression containing variables like \$1, \$2, and so forth, where each numbered variable corresponds to a parenthesized part of the regexp you entered in [Step 5](#). This idMap expression generates the Kerberos ID corresponding to the DN. For example:

\$1@\U\$3.\U\$4

For this example, suppose a DN for your LDAP authentication service is UID=mrslate, OU=Bedrock, DC=slate-quarry, DC=com. Notice that there is no \$2 variable in the idMap example—this is because the second element in the DN, OU=Bedrock, should not be included in the Kerberos ID. The \U modifiers convert the lowercase letters of the domain name (the DC elements) to uppercase.

The two attributes derive the following Kerberos ID from the DN:

```
mrslate@SLATE-QUARRY.COM
```

Step 9 Click **Submit** again to add the second attribute to the User-defined Attributes table.

Importing user accounts from an LDAP authentication service

This section covers authentication services based on the Lightweight Directory Access Protocol (LDAP), including Microsoft Active Directory and the Sun ONE Directory Server (formerly iPlanet).

Users and groups

Imported user accounts use their source directory service (Active Directory or Sun ONE, for example) for authentication, so such accounts always have the same password as they do in the directory service.

When you import a user account, you also import all the groups to which that user belongs. (There is one exception: when you import a member of the Domain Users group from an Active Directory, the Domain Users group is not imported into the grid domain.)

The group memberships of an imported user can become stale over time. For example, suppose you have imported an LDAP user, Fred, who is a member of an LDAP group, Waterbuffaloes. When you import Fred, the Waterbuffaloes group is also imported into the grid. If the LDAP administrator removes Fred from the external Waterbuffaloes group, Fred may remain in the imported Waterbuffaloes group. To keep group memberships up to date, do one of the following:

- Set the authentication service to refresh users on login. See [page 149](#).
- Explicitly reimport either users or groups.

Disabling import on login

By default, LDAP authentication services import users on login—when an unknown user tries to log in to Avaki through an LDAP authentication service, the authentication service attempts to import the user's account from its LDAP directory service.

You can disable the import on login feature, but if you do so, you must explicitly import user accounts from the LDAP directory into the Avaki domain, or import groups that will bring user accounts with them.

- Step 1** To disable the import on login feature, you add an attribute to the Avaki domain's LDAP authentication service. In the data catalog, navigate to
 /System/Domains/<domain-name>/Services/AuthServices/Ldap
 where <domain-name> is the name of your Avaki domain.
- Step 2** Click **Attributes** to display the View Attributes screen. In the User-defined Attributes area, add an attribute with the following properties:
- Name: ldap/importOnDemand
 - Type: String
 - Value: false

If you need more detailed instructions on adding attributes, see [“Creating new attributes” on page 249](#).

Importing the user accounts

To import one or more user accounts into your Avaki domain from an LDAP authentication service, follow these steps. Before you can import users, you must integrate an authentication service into your Avaki domain, as described in [“Integrating LDAP authentication services” on page 148](#).

- Step 1** Log in as a member of the UserAdministrators group.

- Step 2** Navigate to the View Authentication Services screen:

Home > User management > Manage existing authentication services

View Authentication Services

The authentication services below have been integrated into your grid. To view additional information about a service, edit a service's properties, or import users and groups from a service, click on **View/Edit**.

Authentication Service Name	Type	View/Edit Service	Delete Service	Attributes	Security
DefaultAuthService	Grid	View/Edit		Attributes	Security
testnis	NIS	View/Edit	Delete	Attributes	Security
testldap	LDAP	View/Edit	Delete	Attributes	Security

- Step 3** Click **View/Edit** for the authentication service you want to import users from. (If DefaultAuthService is the only authentication service listed, you have not integrated any services yet.) The system displays the LDAP information screen.

- Step 4** Scroll down to the Import Users and Groups area of the screen and click **Import users**.

Import Users and Groups	
Import users	Import users from an LDAP authentication service
Import groups	Import groups from an LDAP authentication service

- Step 5** The system displays the Import LDAP User screen. Click boxes in the Import column to select user accounts to import.
- Step 6** If you don't see the users you want to import, click on a letter, on the **All users** link (both near the top of the screen) or on one of the ranges of numbers near the bottom to display more users. (The ranges—1–25, 26–50, and so on—appear only when a portion of a long list is displayed.)
- Step 7** Click the **Submit** button at the bottom of the screen to bring the user accounts into the grid domain. The system displays a list of the users you have imported.

If you want to set up a home grid directory for a new user account, see [“Setting up home directories” on page 169](#) for instructions.

Importing groups from an LDAP authentication service

This procedure covers all LDAP-based directory services, including Active Directory and the Sun ONE Directory Server (formerly iPlanet).

Before you can import groups, you must integrate an authentication service into your Avaki domain, as described in [“Integrating LDAP authentication services” on page 148](#).

Note When you import a group, you also import all the users who belong to that group. There is one exception: importing the Domain Users group from an Active Directory has no effect—neither the group nor its users are imported into the grid.

To import one or more groups into your Avaki domain from an LDAP authentication service, follow the steps below.

- Step 1** Log in as a member of the UserAdministrators group.

Step 2 Navigate to the View Authentication Services screen:

Home > User management > Manage existing authentication services

View Authentication Services

The authentication services below have been integrated into your grid. To view additional information about a service, edit a service's properties, or import users and groups from a service, click on **View/Edit**.

Authentication Service Name	Type	View/Edit Service	Delete Service	Attributes	Security
DefaultAuthService	Grid	View/Edit		Attributes	Security
testnis	NIS	View/Edit	Delete	Attributes	Security
testldap	LDAP	View/Edit	Delete	Attributes	Security

Step 3 Click **View/Edit** for the authentication service you want to import groups from. (If DefaultAuthService is the only authentication service listed, you have not integrated any services yet.) The LDAP Information screen appears.

Step 4 Scroll down to the Import Users and Groups area of the screen and click **Import groups**.

Import Users and Groups

Import users	Import users from an LDAP authentication service
Import groups	Import groups from an LDAP authentication service

Step 5 The system displays the Import LDAP Group screen. Click boxes in the Import column to select groups to import.

Step 6 Click **Submit**. The system displays the Import LDAP Group screen.

Import LDAP Group

To import group accounts from an LDAP authentication service into the data grid, click boxes in the Import column to select the group accounts to import, then click **Submit**.

Note: When you import a group, you import all users in the group and all groups to which those users belong.

[All groups](#)

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Import	Group name
<input type="checkbox"/>	Account Operators
<input type="checkbox"/>	Accounting
<input type="checkbox"/>	Administrators
<input type="checkbox"/>	Services
<input type="checkbox"/>	Techies

Import all groups in LDAP authentication service.

Step 7 If you don't see the groups you want to import, click on a letter, on the **All groups** link (both near the top of the screen) or on one of the ranges of numbers near the bottom to display more groups. (The ranges—1–25, 26–50, and so on—appear only when a portion of a long list is displayed.)

Step 8 Click boxes in the Import column to select the groups you want to import.

Step 9 Click the **Import** button at the bottom of the screen to bring the groups into the Avaki domain. The system displays a list of the groups you have imported.

Integrating NIS authentication services

To integrate a Network Information Service (NIS) directory service into your Avaki domain and create a corresponding authentication service in the domain, follow these steps.

Step 1 Log in to Avaki as a member of the Administrators group.

Step 2 Navigate to the Integrate NIS Authentication Service screen:

Home > User management > Integrate NIS authentication service

Integrate NIS Authentication Service

For "Service name," specify a name for the NIS authentication service. This service name, which can be any combination of letters or numbers, will appear in the list of authentication services that users can log in to.

Service name:

NIS machine (DNS name or IP address):

NIS domain:

Refresh users on login:

Step 3 Fill in the form:

- Service name: Devise a name for the new service and enter it here.
- NIS machine (DNS name or IP address): Specify the name or IP address of the host machine on which the NIS service resides.
- NIS domain: Enter the name of the NIS domain for the authentication service you want to integrate.
- Refresh users on login: If you check this box, each user account imported through this authentication service will be reimported every time the user logs in. Checking this box is a good way to ensure that changes to an imported user account, such as group membership changes, are available to Avaki. If you do not check this box, we recommend that you periodically reimport user accounts manually—see [“Keeping authentication services up to date”](#) on page 185 for more information.

If you're not sure what to enter in the NIS machine or NIS domain fields, ask your NIS administrator.

Step 4 Click **Submit** to integrate the authentication service. The system displays a summary of the information you entered.

For instructions on importing users and groups from a newly integrated authentication service, see “[Importing user accounts from an NIS authentication service](#),” below, and “[Importing groups from an NIS authentication service](#)” on page 165.

Importing user accounts from an NIS authentication service

Users and groups

Imported user accounts use their source NIS directory service for authentication, so such accounts always have the same password as they do in the directory service.

When you import a user account, you also import all the groups to which that user belongs.

The group memberships of an imported user can become stale over time. For example, suppose you have imported an NIS user, Fred, who is a member of an NIS group, Waterbuffaloes. (When you import Fred, the Waterbuffaloes group is also imported.) The NIS administrator may remove Fred from the original Waterbuffaloes group, but he will remain in the imported Waterbuffaloes group in the grid. To keep group memberships up to date, do one of the following:

- Set the authentication service to refresh users on login. See [page 162](#).
- Explicitly reimport users or groups.

Disabling import on login

By default, NIS authentication services import users on login—when an unknown user tries to log in to Avaki through an NIS authentication service, the authentication service attempts to import the user’s account from its NIS directory service.

You can disable the import on login feature, but if you do so, you must explicitly import user accounts from the NIS directory into the Avaki domain, or import groups that will bring user accounts with them.

- Step 1** To disable the import on login feature, you add an attribute to the Avaki domain’s NIS authentication service. In the data catalog, navigate to

```
/System/Domains/<domain-name>/Services/AuthServices/Nis
```

where <domain-name> is the name of your Avaki domain.

Step 2 Click **Attributes** to display the View Attributes screen. In the User-defined Attributes area, add an attribute with the following properties:

- Name: nis/importOnDemand
- Type: String
- Value: false

If you need more detailed instructions on adding attributes, see [“Creating new attributes” on page 249](#).

Importing the user accounts

To import one or more user accounts into your grid domain from a Network Information Service (NIS) authentication service, follow these steps. Before you can import users, you must integrate an authentication service into your Avaki domain, as described in [“Integrating NIS authentication services” on page 162](#).

Step 1 Log in as a member of the UserAdministrators group.

Step 2 Navigate to the View Authentication Services screen:

Home > User management > Manage existing authentication services

View Authentication Services

The authentication services below have been integrated into your grid. To view additional information about a service, edit a service's properties, or import users and groups from a service, click on **View/Edit**.

Authentication Service Name	Type	View/Edit Service	Delete Service	Attributes	Security
DefaultAuthService	Grid	View/Edit		Attributes	Security
testnis	NIS	View/Edit	Delete	Attributes	Security
testldap	LDAP	View/Edit	Delete	Attributes	Security

Step 3 Click **View/Edit** for the authentication service you want to import users from. (If DefaultAuthService is the only authentication service listed, you have not integrated any services yet.)

Step 4 In the Import Users and Groups box, click **Import Users**. The system displays the Import NIS User screen.

Step 5 Click boxes in the Import column to select user accounts to import.

Step 6 Click the **Submit** button at the bottom of the screen to bring the user accounts into the grid domain. The system displays a list of the users you have imported.

If you want to set up a home directory for a new user account, see “[Setting up home directories](#)” on page 169 for instructions.

Importing groups from an NIS authentication service

To import one or more groups into your Avaki domain from an NIS authentication service, follow these steps. Before you can import groups, you must integrate an authentication service into your Avaki domain, as described in “[Integrating NIS authentication services](#)” on page 162.

Note When you import a group, you also import all the users who belong to that group.

Step 1 Log in as a member of the UserAdministrators group.

Step 2 Navigate to the View Authentication Services screen:

Home > User management > Manage existing authentication services

View Authentication Services

The authentication services below have been integrated into your grid. To view additional information about a service, edit a service's properties, or import users and groups from a service, click on **View/Edit**.

Authentication Service Name	Type	View/Edit Service	Delete Service	Attributes	Security
DefaultAuthService	Grid	View/Edit		Attributes	Security
testnis	NIS	View/Edit	Delete	Attributes	Security
testldap	LDAP	View/Edit	Delete	Attributes	Security

Step 3 Click **View/Edit** for the authentication service you want to import groups from. (If DefaultAuthService is the only authentication service listed, you have not integrated any services yet.)

Step 4 The NIS Information screen appears. Scroll down to the Import Users and Groups box and click **Import Groups**. The system displays the Import NIS Group screen.

Step 5 If you don't see the groups you want to import, click on a letter, on the **All groups** link (both near the top of the screen) or on **Next 25** (near the bottom) to display more users.

Step 6 Click boxes in the Import column to select the groups you want to import.

Step 7 Click the **Submit** button at the bottom of the screen to bring the groups into the Avaki domain. The system displays a list of the groups you have imported.

Viewing existing authentication services

To display information on authentication services in your Avaki domain, follow the steps below. This procedure applies to authentication services of all types (LDAP, NIS, and grid).

Step 1 Log in as a member of the UserAdministrators group.

Step 2 Navigate to the View Authentication Services screen:

Home > User management > View existing authentication services

View Authentication Services

The authentication services below have been integrated into your grid. To view additional information about a service, edit a service's properties, or import users and groups from a service, click on **View/Edit**.

Authentication Service Name	Type	View/Edit Service	Delete Service	Attributes	Security
DefaultAuthService	Grid	View/Edit		Attributes	Security
testnis	NIS	View/Edit	Delete	Attributes	Security
testldap	LDAP	View/Edit	Delete	Attributes	Security

Step 3 Click **View/Edit** for the authentication service you're interested in. The system displays the information for the service.

Deleting authentication services

To remove an authentication service from your Avaki domain, follow the steps below. This procedure applies to LDAP and NIS authentication services.

Caution When you remove an authentication service, any users and groups imported from that service are also removed from the Avaki domain.

Step 1 Log in as a member of the Administrators group.

Step 2 Navigate to the View Authentication Services screen:

Home > User management > View existing authentication services

View Authentication Services

The authentication services below have been integrated into your grid. To view additional information about a service, edit a service's properties, or import users and groups from a service, click on **View/Edit**.

Authentication Service Name	Type	View/Edit Service	Delete Service	Attributes	Security
DefaultAuthService	Grid	View/Edit		Attributes	Security
testnis	NIS	View/Edit	Delete	Attributes	Security
testldap	LDAP	View/Edit	Delete	Attributes	Security

- Step 3** Click the **Delete** link for the authentication service you want to delete. The system deletes the authentication service and removes it from the screen.

Managing users

Avaki grids support two types of user accounts:

- Grid accounts, which you create in Avaki software.
Use grid accounts for users who will be assigned to grid-specific groups—groups that, under your organization's security policies, might not be allowed to exist outside the grid.
- Imported accounts, which you bring into Avaki from an external directory service. Importing user accounts and related issues are discussed in the section [“Managing authentication services” on page 148](#).

This section includes procedures covering both types of accounts:

- [“Creating grid user accounts,”](#) below
- [“Setting up home directories” on page 169](#)
- [“Changing a grid user's password” on page 175](#)
- [“Mapping Avaki users to database users” on page 176](#)
- [“Keeping authentication services up to date” on page 185](#)
- [“Viewing and modifying user account information” on page 188](#)
- [“Deleting user accounts” on page 190](#)

Creating grid user accounts

To create an account, follow these steps.

Step 1 Log in as a member of the UserAdministrators group.

Step 2 Navigate to the Create Grid User screen:

Home > User management > Create users

The screenshot shows a web form titled "Create Grid User". It contains five text input fields stacked vertically, each with a label to its left: "Username:", "Full name:", "Email:", "Password:", and "Confirm password:". At the bottom of the form, there are two buttons: "Submit" and "Cancel".

Step 3 Enter a name for the user account, the user's actual first and last names, and other information required on the form.

Step 4 Click **Submit**. The system displays a summary of the information you entered for the new account.

Step 5 Click **Done** to return to the Data Grid User Management screen.

Step 6 If this user needs special privileges, add the user to a privileged grid group such as DataProviders or DatabaseAdministrators. See [“Adding users to groups” on page 192](#) for instructions. (For more information on DataProviders and other privileged grid groups, see the *Sybase Avaki EII Overture*.)

If you wish to set up a home directory for the new user account, see [“Setting up home directories,”](#) below, for instructions.

Setting up home directories

Home directories for users are not set up automatically when you create or import a user account. If you provide a home directory, the user is not moved there upon logging in; the home directory is just a place to store files and subdirectories.

If you want a new user to have a home directory in the data catalog, do one of the following:

- Share the user's local home directory into the grid. See [“Creating Avaki shares” on page 258](#) for instructions on creating Avaki shares. Be sure to designate the user as the new owner of the share and set the permissions so that the user has control of the shared directory and its contents. This is the preferred option.
- Create an Avaki directory as described in [“Creating the home directory,”](#) below. If you choose this option, the user will not be able to share files into the home directory unless he or she is a member of the DataProviders or Administrators groups. (Only members of those groups can create Avaki shares.)

Creating the home directory

Follow these steps to make a directory for a new user.








- Step 1** Log in as a member of the UserAdministrators group.
- Step 2** Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>	 /	Attributes	Security
<input type="checkbox"/>	 GeneratedViews	Attributes	Security
<input type="checkbox"/>	 Interconnects	Attributes	Security
<input type="checkbox"/>	 Metadata	Attributes	Security
<input type="checkbox"/>	 Shares	Attributes	Security
<input type="checkbox"/>	 System	Attributes	Security
<input type="checkbox"/>	 WSDLs	Attributes	Security

Step 3 Navigate to the Avaki directory in which you want to create the home directory.

Step 4 Click **Create directory**. The system displays the Create New Directory screen.

Create New Directory

Parent directory: /

New directory name:

Step 5 Enter a name for the new directory, such as the name of the user account.

Step 6 Click **Submit**. The system displays the Browse Directories screen showing the new directory.

Step 7 Configure the directory's permissions as described in [“Setting ACLs on a grid object,”](#) below.

Setting ACLs on a grid object

Follow these steps to configure the new Avaki directory so that the user owns it and has read, write, execute and delete permissions for the directory and its contents. (The procedure for setting ACLs on other objects, such as files, is essentially the same.)

Step 1 Log in as a member of the UserAdministrators group.








Step 2 Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>		/	Attributes	Security
<input type="checkbox"/>		GeneratedViews	Attributes	Security
<input type="checkbox"/>		Interconnects	Attributes	Security
<input type="checkbox"/>		Metadata	Attributes	Security
<input type="checkbox"/>		Shares	Attributes	Security
<input type="checkbox"/>		System	Attributes	Security
<input type="checkbox"/>		WSDLs	Attributes	Security

Step 3 Click on directory names to navigate to the new home directory.

- Step 4** Click the **Security** link for the new home directory. The system displays the View Security Information screen.

View Security Information

You are viewing security information for the following object:
/home/fred

The current owner of the object is the user **Administrator** in domain **Cambridge** and the authentication service **DefaultAuthService**.

The following users and groups have been added to the Access Control List (ACL) for this object. To modify a user or group's permissions, place a check mark next to the user or group and click **Edit All Checked**. To add a user who is in the current grid domain, click **Add User to ACL**. To add a group that is in the current domain, click **Add Group to ACL**. To add a user or group that is in a connected domain, click **Add Via Interconnect ID**.

Select	Name	Type	Domain	Auth Service	Read	Write	Execute	Delete
<input type="checkbox"/>	DomainUsers	group	Cambridge	DefaultAuthService	allow	unset	unset	unset
<input type="checkbox"/>	Administrator	user	Cambridge	DefaultAuthService	allow	allow	allow	allow

Manage Access Control Lists

[Edit All Checked](#) Configure permissions for each user or group in the ACL for an object

[Add User to ACL](#) Add a user to the ACL for an object

[Add Group to ACL](#) Add a group to the ACL for an object

[Add Via Interconnect ID](#) Upload an interconnect ID to add a user or group to the ACL for an object

- Step 5** Click **Add User to ACL** at the bottom of the screen. The Add Grid User to ACL screen appears.

Add Grid User to ACL

All users
 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

In the Add column, place a check mark next to the names of the user(s) you want to add to the access control list (ACL) for the following object:

/home/fred

Add	Username	Authentication service
<input type="checkbox"/>	Administrator	DefaultAuthService
<input type="checkbox"/>	ed_test	testldap
<input type="checkbox"/>	fred	DefaultAuthService
<input type="checkbox"/>	jtestuser	testnis
<input type="checkbox"/>	rip	testldap
<input type="checkbox"/>	wilma	DefaultAuthService

- Step 6** In the Add column, click the check box for the user to whom the home directory belongs, then click **Submit**. The system displays the Modify Permissions screen for the home directory.

Modify Permissions

You may specify read, write, execute, or delete permission for each user or group in the ACL for for the following object:

/home/fred

For each permission, you may specify the allow, deny, or unset option, or you may leave the current value as is.

Select the new permissions for the user or group:

Name	Type	Domain	Auth Service	Set as Owner	Permissions															
fred	user	Cambridge	DefaultAuthService	<input type="checkbox"/>	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;"></th> <th style="width: 15%;">Current</th> <th style="width: 70%;">New</th> </tr> </thead> <tbody> <tr> <td>Read</td> <td>unset</td> <td>as is ▾</td> </tr> <tr> <td>Write</td> <td>unset</td> <td>as is ▾</td> </tr> <tr> <td>Execute</td> <td>unset</td> <td>as is ▾</td> </tr> <tr> <td>Delete</td> <td>unset</td> <td>as is ▾</td> </tr> </tbody> </table>		Current	New	Read	unset	as is ▾	Write	unset	as is ▾	Execute	unset	as is ▾	Delete	unset	as is ▾
	Current	New																		
Read	unset	as is ▾																		
Write	unset	as is ▾																		
Execute	unset	as is ▾																		
Delete	unset	as is ▾																		

Apply changes to selected directory only
 Apply changes to selected directory and all contents

- Step 7** Click the Set as Owner check box for the new user.
- Step 8** In the New part of the Permissions column (in the new user's row), use the pull-down menus to set Read, Write, Execute, and Delete to Allow.
- Step 9** Click **Apply changes to selected object and all contents**. This makes your changes recursive.
- Step 10** Click **Submit** to change the home directory's ownership and permissions. The View Security Information screen reappears.

If you want to change the permissions on this directory for the Administrator or for members of the DomainUsers group, select them and click **Edit all checked**.

Changing a grid user's password

To change the password on a grid user account, follow these steps.

Note You cannot use Avaki software to change passwords for users imported from an external authentication service.

Step 1 Log in as a member of the UserAdministrators group.

Step 2 Navigate to the Select User screen:

Home > User management > View and modify users

Select User

To modify a user's personal information, download a user interconnect ID, enable or disable the user (auth service type Grid only) or change a password, select the user and click the **View/Edit** button. To delete a user from a grid domain, select the user and click **Delete All Checked**. To configure the user ID (UID) that is mapped to a user account, select the user and click **Update UIDs** below.

If you have a two-way connection with a remote domain, you may want to enable the administrator of the remote domain to view users in your domain and add them to groups in the remote domain. To do this, you must give the administrator permission to view and edit those users, as follows:

1. Obtain the interconnect ID that represents the administrator of the remote domain.
2. In the table below, select the users whose settings you want to allow the remote administrator to modify.
3. Go to the [Expose Users](#) screen to continue.

Note: The remote administrator will be able to edit only the users you selected, but the administrator will be able to view *all* the users in your domain.

All users

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Select	Username	Auth Service	Auth Service Type	Enabled	View/Edit User	Attributes	Security
<input type="checkbox"/>	Administrator	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	MessagingUser	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	barney	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	betty	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	fred	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	wilma	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security

Total number of users in domain: 6

Step 3 If you don't see the user account you're looking for, click on one of the letters or on the **All Users** link to display more users.

- Step 4** In the View/Edit User column, click **View/Edit** for the user whose password you want to change. The system displays a screen with a Change Password area.
- Step 5** Enter the new password in both the New password and Confirm new password fields.
- Step 6** Click **Submit** to change the password. The system displays a confirmation screen.

Mapping Avaki users to database users

Avaki database connectors are typically configured with a single default database account, which is used to execute every query.

However, database connectors can be configured to allow mappings between Avaki user accounts and database user accounts. This feature, which is called database identity mapping, makes it possible for an Avaki database operation to use different credentials when it connects to the database and thus to retrieve information whose access is restricted to particular database users. The mapping feature also means that when two users execute the same database operation, they might (depending on how the database permissions are configured) retrieve different results.

The mappings can be set up and maintained either by grid administrators or by Avaki users themselves. (By “Avaki users,” we mean both grid users and imported users.)

Here’s an example: a user called Betty has permission to retrieve monthly sales figures from a database called NortheastSales. Betty wants to use Avaki to combine these figures with result from other databases, so she does the following:

- Asks a member of the DatabaseAdministrators group to set up a database connector, Dbconn1, for the NortheastSales database and to enable database identity mapping on Dbconn1.
- Asks a member of the DatabaseAdministrators group to set up a database operation that uses Dbconn1 to query the NortheastSales database for the monthly sales numbers.
- Sets up a mapping between her Avaki user ID and her database user ID. (See [“Adding database identity mappings” on page 177](#) for instructions on setting up database ID mappings.)

When Betty is logged in to the Avaki domain, she can execute the database operation (or a data service or view generator that calls the database operation), and Avaki will use her database user account and password when it queries the NortheastSales database and gets the sales figures. Any other Avaki user executing the same database operation will be unable to retrieve the sales figures unless a similar mapping has been

configured to a database user account with appropriate permissions. If Betty (or another user) executes a database operation on a database connector where identity mapping is not enabled, she will be able to retrieve only results that are accessible to the default user for that database connector.

Caution In Avaki domains where all servers are running release 6.0 or later, no results are ever cached for users with mapped database identities. However, if you run a mapped database operation that uses a cache service on an older (release 5.x) grid server, the results may be cached—and once cached, the results will be available to other users of that database operation. So, in the case of our example, users other than Betty might be able to see confidential sales figures. Consequently, we strongly recommend that you avoid using database identity mapping in Avaki domains where any grid server is running a release older than 6.0.

The rest of this section explains how to configure database identity mappings:

- [“Adding database identity mappings,”](#) below
- [“Viewing database identity mappings” on page 180](#)
- [“Modifying database identity mappings” on page 182](#)
- [“Deleting database identity mappings” on page 183](#)

For information on enabling database identity mapping on database connectors, see the *Sybase Avaki EII Provisioning and Advanced Data Integration Guide*.

Adding database identity mappings

To set up mappings between Avaki and database user IDs, follow these steps. You must be logged in as either the Avaki user whose account you are editing or as a member of the UserAdministrators group.

Note If no database connectors exist, you can’t complete this procedure.

Step 1 Navigate to the Select User screen:

Home > User management > View and modify users

Select User

To modify a user's personal information, download a user interconnect ID, enable or disable the user (auth service type Grid only) or change a password, select the user and click the **View/Edit** button. To delete a user from a grid domain, select the user and click **Delete All Checked**. To configure the user ID (UID) that is mapped to a user account, select the user and click **Update UIDs** below.

If you have a two-way connection with a remote domain, you may want to enable the administrator of the remote domain to view users in your domain and add them to groups in the remote domain. To do this, you must give the administrator permission to view and edit those users, as follows:

1. Obtain the interconnect ID that represents the administrator of the remote domain.
2. In the table below, select the users whose settings you want to allow the remote administrator to modify.
3. Go to the [Expose Users](#) screen to continue.

Note: The remote administrator will be able to edit only the users you selected, but the administrator will be able to view *all* the users in your domain.

All users

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Select	Username	Auth Service	Auth Service Type	Enabled	View/Edit User	Attributes	Security
<input type="checkbox"/>	Administrator	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	MessagingUser	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	barney	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	betty	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	fred	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	wilma	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security

Total number of users in domain: 6

Step 2 If you don't see the user account you're looking for, click one of the letters to display a list of users whose names begin with that letter.

Step 3 Click on **View/Edit** for the user account you want to modify. The system displays the View/Modify User Info screen.

- Step 4** In the Database Identity Mappings section, click **View/Edit Mappings**. The View/Edit Identity Mappings screen appears.

View/Edit Identity Mappings			
You are viewing the database identity mappings for betty.			
Select	Database Connector	Mapped Username	View/Edit
There are no identity mappings at present.			
<input type="button" value="Check All"/>		<input type="button" value="Clear All"/>	
<input type="button" value="Delete All Checked"/>		<input type="button" value="Add Mapping"/>	<input type="button" value="Done"/>

- Step 5** Click **Add Mapping**. The Select Database Connector screen appears.

Select Database Connector	
Grid domain:	<input type="text" value="Bedrock"/>
You have chosen to create a new database connector identity mapping. Select the database connector to use.	
Select	Connector Name
<input type="radio"/>	NortheastSales
<input type="button" value="Continue"/> <input type="button" value="Cancel"/>	

- Step 6** Click a button in the Select column to choose the database connector for which this mapping will be in effect. Then click **Continue**. The View/Edit Identity Mapping screen appears.

View/Edit Identity Mapping	
To create or update this database identity mapping, please enter the information below and click Continue .	
Database connector:	Bedrock.NortheastSales
Mapped username:	<input type="text"/>
(optional) Password:	<input type="text"/>
(optional) Confirm Password:	<input type="text"/>
<input type="button" value="Continue"/> <input type="button" value="Cancel"/>	

- Step 7** Fill in the form:

- **Mapped username:** Enter the user ID for the database to which the specified database connector connects.

- Password: Enter the password for the database user ID above.
- Confirm password: Enter the password again.

Step 8 Click **Continue**. The View/Edit Identity Mappings screen reappears, now showing the user name you just mapped.

Viewing database identity mappings

To display mappings between Avaki and database user IDs for an Avaki user account, follow these steps. You must be logged in as either the Avaki user whose account you are editing or as a member of the UserAdministrators group.

Step 1 Navigate to the Select User screen:

Home > User management > View and modify users

Select User

To modify a user's personal information, download a user interconnect ID, enable or disable the user (auth service type Grid only) or change a password, select the user and click the **View/Edit** button. To delete a user from a grid domain, select the user and click **Delete All Checked**. To configure the user ID (UID) that is mapped to a user account, select the user and click **Update UIDs** below.

If you have a two-way connection with a remote domain, you may want to enable the administrator of the remote domain to view users in your domain and add them to groups in the remote domain. To do this, you must give the administrator permission to view and edit those users, as follows:

1. Obtain the interconnect ID that represents the administrator of the remote domain.
2. In the table below, select the users whose settings you want to allow the remote administrator to modify.
3. Go to the [Expose Users](#) screen to continue.

Note: The remote administrator will be able to edit only the users you selected, but the administrator will be able to view *all* the users in your domain.

All users

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Select	Username	Auth Service	Auth Service Type	Enabled	View/Edit User	Attributes	Security
<input type="checkbox"/>	Administrator	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	MessagingUser	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	barney	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	betty	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	fred	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	wilma	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security

Total number of users in domain: 6

- Step 2** If you don't see the user account you're looking for, click one of the letters to display a list of users whose names begin with that letter.
- Step 3** Click on **View/Edit** for the user account you want to modify. The system displays the View/Modify User Info screen.
- Step 4** In the Database Identity Mappings section, click **View/Edit Mappings**. The View/Edit Identity Mappings screen appears.

View/Edit Identity Mappings

You are viewing the database identity mappings for betty.

Select	Database Connector	Mapped Username	View/Edit
<input type="checkbox"/>	Bedrock.NortheastSales	brubble	View/Edit

Modifying database identity mappings

To set up mappings between Avaki and database user IDs, follow these steps. You must be logged in as either the Avaki user whose account you are editing or as a member of the UserAdministrators group.

Step 1 Navigate to the Select User screen:

Home > User management > View and modify users

Select User

To modify a user's personal information, download a user interconnect ID, enable or disable the user (auth service type Grid only) or change a password, select the user and click the **View/Edit** button. To delete a user from a grid domain, select the user and click **Delete All Checked**. To configure the user ID (UID) that is mapped to a user account, select the user and click **Update UIDs** below.

If you have a two-way connection with a remote domain, you may want to enable the administrator of the remote domain to view users in your domain and add them to groups in the remote domain. To do this, you must give the administrator permission to view and edit those users, as follows:

1. Obtain the interconnect ID that represents the administrator of the remote domain.
2. In the table below, select the users whose settings you want to allow the remote administrator to modify.
3. Go to the [Expose Users](#) screen to continue.

Note: The remote administrator will be able to edit only the users you selected, but the administrator will be able to view *all* the users in your domain.

All users

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Select	Username	Auth Service	Auth Service Type	Enabled	View/Edit User	Attributes	Security
<input type="checkbox"/>	Administrator	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	MessagingUser	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	barney	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	betty	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	fred	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	wilma	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security

Total number of users in domain: 6

Step 2 If you don't see the user account you're looking for, click one of the letters to display a list of users whose names begin with that letter.

Step 3 Click **View/Edit** for the user account you want to modify. The system displays the View/Modify User Info screen.

- Step 4** In the Database Identity Mappings section, click **View/Edit Mappings**. The View/Edit Identity Mappings screen appears.

View/Edit Identity Mappings			
You are viewing the database identity mappings for betty.			
Select	Database Connector	Mapped Username	View/Edit
<input type="checkbox"/>	Bedrock.NortheastSales	brubble	View/Edit
<input type="button" value="Check All"/> <input type="button" value="Clear All"/>			
<input type="button" value="Delete All Checked"/> <input type="button" value="Add Mapping"/> <input type="button" value="Done"/>			

- Step 5** Click **View/Edit** for the mapping you want to modify. The system displays the View/Edit Identity Mapping screen.

View/Edit Identity Mapping	
To create or update this database identity mapping, please enter the information below and click Continue .	
Database connector:	Bedrock.NortheastSales
Mapped username:	<input type="text"/>
(optional) Password:	<input type="text"/>
(optional) Confirm Password:	<input type="text"/>
<input type="button" value="Continue"/> <input type="button" value="Cancel"/>	

- Step 6** Enter a new user name for the database account or a new password, then click **Continue**. The View/Edit Identity Mappings screen reappears.

Deleting database identity mappings

Follow these steps to remove one or more mappings between Avaki and database user IDs. You must be logged in as either the Avaki user whose account you are editing or as a member of the UserAdministrators group.

Note An alternative to deleting individual mappings is to disable database identity mapping on the database connector. This change affects all users of all database operations associated with the database connector. Do the following:

- Navigate to Home > Database provisioning > Manage database connectors.
- Click **View/Edit** for the database connector you want to modify.
- Uncheck the “Allow database identity mappings” box.

- Click **Submit** to save the change.

Step 1 Navigate to the Select User screen:

Home > User management > View and modify users

Select User

To modify a user's personal information, download a user interconnect ID, enable or disable the user (auth service type Grid only) or change a password, select the user and click the **View/Edit** button. To delete a user from a grid domain, select the user and click **Delete All Checked**. To configure the user ID (UID) that is mapped to a user account, select the user and click **Update UIDs** below.

If you have a two-way connection with a remote domain, you may want to enable the administrator of the remote domain to view users in your domain and add them to groups in the remote domain. To do this, you must give the administrator permission to view and edit those users, as follows:

1. Obtain the interconnect ID that represents the administrator of the remote domain.
2. In the table below, select the users whose settings you want to allow the remote administrator to modify.
3. Go to the [Expose Users](#) screen to continue.

Note: The remote administrator will be able to edit only the users you selected, but the administrator will be able to view *all* the users in your domain.

All users

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Select	Username	Auth Service	Auth Service Type	Enabled	View/Edit User	Attributes	Security
<input type="checkbox"/>	Administrator	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	MessagingUser	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	barney	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	betty	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	fred	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	wilma	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security

Total number of users in domain: 6

Step 2 If you don't see the user account you're looking for, click one of the letters to display a list of users whose names begin with that letter.

Step 3 Click on **View/Edit** for the user account you want to modify. The system displays the View/Modify User Info screen.

- Step 4** In the Database Identity Mappings section, click **View/Edit Mappings**. The View/Edit Identity Mappings screen appears.

View/Edit Identity Mappings

You are viewing the database identity mappings for betty.

Select	Database Connector	Mapped Username	View/Edit
<input type="checkbox"/>	Bedrock.NortheastSales	brubble	View/Edit

- Step 5** Click a box in the Select column for the mapping you want to delete, then click **Delete All Checked**. (You can select and delete multiple mappings at a time.) Confirm the deletion when you're prompted to do so.

Keeping authentication services up to date

Information in authentication services that has been imported into an Avaki domain from an external directory service can become outdated in several ways, including the following:

- A user's group memberships change
- New users are added

There are three ways to keep an LDAP authentication service up to date:

Update method	Description	For instructions, see...
Configure scheduled refreshes	Imports new users and refreshes accounts that have already been imported	“Configuring LDAP authentication services to refresh on a schedule” on page 186
Set the authentication service to refresh users on login	Refreshes only user accounts that have already been imported	page 149 —you configure refresh on login when you import the authentication service
Manually re-import user accounts	Do this whenever group memberships change in the external directory service. Necessary only if you choose not to use either of the other methods.	“Importing user accounts from an LDAP authentication service” on page 157

Configuring LDAP authentication services to refresh on a schedule

There are four types of schedules: one-time, periodic, calendared, and advanced. You configure one-time, periodic, and calendared schedules using the UI's graphical tools. An advanced schedule is similar to a calendared schedule, except that in an advanced schedule you must supply a cron expression to specify the recurrence interval. Cron expressions provide more flexibility in specifying times. For example, they allow you to schedule to the precise minute and second (the UI offers 5-minute granularity), and they allow you to schedule two or more times in one schedule entry (such as 5 am and 6 pm every Tuesday) where the UI allows only one. For information on the syntax of cron expressions, see the *Sybase Avaki EII Command Reference*.

Follow these steps to add entries to the refresh schedule for an LDAP authentication service. You can create multiple entries for one authentication service if you wish.

- Step 1** To display a list of LDAP authentication service in this Avaki domain, navigate to the View Authentication Services screen:

Home > User management > Manage existing authentication services

View Authentication Services

The authentication services below have been integrated into your grid. To view additional information about a service, edit a service's properties, or import users and groups from a service, click on **View/Edit**.

Authentication Service Name	Type	View/Edit Service	Schedule	Delete Service	Attributes	Security
DefaultAuthService	Grid	View/Edit			Attributes	Security
testldap	LDAP	View/Edit	Schedule	Delete	Attributes	Security

- Step 2** Click the **Schedule** link for the authentication service you want to modify.

- Step 3** On the screen that appears, click **Add schedule**. The system displays the Add New Schedule screen.

Add New Schedule

Select a time zone for this schedule: (server's local time zone) ▾

Type of schedule:

One time
 Periodic
 Calendared
 Advanced

Starting Now
 12 AM ▾ :00 ▾ Jan ▾ 1 ▾ 2005 ▾

Recur every: minutes ▾

Continue recurring

forever

at most times

until 12 AM ▾ :00 ▾ Jan ▾ 1 ▾ 2005 ▾

Schedule exclusions:

No schedule exclusions are currently defined in this domain.

- Step 4** Click a tab to choose the type of schedule you want: One time, Periodic, Calendared, or Advanced.

- Step 5** Go to the appropriate procedure to complete your schedule entry:

- [“Configuring one-time refresh schedules” on page 269](#)
- [“Configuring periodic refresh schedules” on page 270](#)
- [“Configuring calendared refresh schedules” on page 271](#)
- [“Configuring advanced refresh schedules” on page 273](#)

Viewing and modifying user account information

To display and change the information associated with grid user accounts, follow these steps.

- Step 1** Log in as a member of the UserAdministrators group.
- Step 2** Navigate to the Select User screen:

Home > User management > View and modify users

Select User

To modify a user's personal information, download a user interconnect ID, enable or disable the user (auth service type Grid only) or change a password, select the user and click the **View/Edit** button. To delete a user from a grid domain, select the user and click **Delete All Checked**. To configure the user ID (UID) that is mapped to a user account, select the user and click **Update UIDs** below.

If you have a two-way connection with a remote domain, you may want to enable the administrator of the remote domain to view users in your domain and add them to groups in the remote domain. To do this, you must give the administrator permission to view and edit those users, as follows:

1. Obtain the interconnect ID that represents the administrator of the remote domain.
2. In the table below, select the users whose settings you want to allow the remote administrator to modify.
3. Go to the [Expose Users](#) screen to continue.

Note: The remote administrator will be able to edit only the users you selected, but the administrator will be able to view *all* the users in your domain.

All users

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

Select	Username	Auth Service	Auth Service Type	Enabled	View/Edit User	Attributes	Security
<input type="checkbox"/>	Administrator	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	MessagingUser	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	barney	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	betty	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	fred	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	wilma	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security

Total number of users in domain: 6

- Step 3** If you don't see the user account you're looking for, click on one of the letters or on the **All Users** link to display more users.

- Step 4** Click on **View/Edit** for the user account you want to review or change. The system displays the Modify User Info screen.

View/Modify User Info					
Username: betty					
Full name:	<input type="text" value="Betty Rubble"/>				
Email:	<input type="text" value="betty@bedrock.net"/>				
UID for NFS: unknown					
Enabled:	<input checked="" type="checkbox"/>				
Groups:	<table border="1"> <thead> <tr> <th>Group Name</th> <th>Domain</th> </tr> </thead> <tbody> <tr> <td>DomainUsers</td> <td>Bedrock</td> </tr> </tbody> </table>	Group Name	Domain	DomainUsers	Bedrock
Group Name	Domain				
DomainUsers	Bedrock				
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>					

Download User Interconnect ID
To send the user's interconnection ID to a remote domain, click Download Interconnect ID .
Download Interconnect ID

Database Identity Mappings
To view or edit the user's database identity mappings, click View/Edit Mappings .
View/Edit Mappings

Change Password
To change the user's password, enter the new password in the "New password" and "Confirm new password" fields.
New password: <input type="text"/>
Confirm new password: <input type="text"/>
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>

- Step 5** Edit the user info as needed.
- Step 6** Click **Submit** to save your changes. The system displays the updated user information.

Deleting user accounts

To delete user accounts from an Avaki domain, follow these steps.

Step 1 Log in as a member of the UserAdministrators group.

Step 2 Navigate to the Select User screen:

Home > User management > View and modify users

Select User

To modify a user's personal information, download a user interconnect ID, enable or disable the user (auth service type Grid only) or change a password, select the user and click the **View/Edit** button. To delete a user from a grid domain, select the user and click **Delete All Checked**. To configure the user ID (UID) that is mapped to a user account, select the user and click **Update UIDs** below.

If you have a two-way connection with a remote domain, you may want to enable the administrator of the remote domain to view users in your domain and add them to groups in the remote domain. To do this, you must give the administrator permission to view and edit those users, as follows:

1. Obtain the interconnect ID that represents the administrator of the remote domain.
2. In the table below, select the users whose settings you want to allow the remote administrator to modify.
3. Go to the [Expose Users](#) screen to continue.

Note: The remote administrator will be able to edit only the users you selected, but the administrator will be able to view *all* the users in your domain.

All users

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

Select	Username	Auth Service	Auth Service Type	Enabled	View/Edit User	Attributes	Security
<input type="checkbox"/>	Administrator	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	MessagingUser	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	barney	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	betty	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	fred	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security
<input type="checkbox"/>	wilma	DefaultAuthService	Grid	Enabled	View/Edit	Attributes	Security

Total number of users in domain: 6

Step 3 If you don't see the users you want to remove, click on a letter or on the **All users** link to display more users.

Step 4 Click boxes in the Select column to choose the user accounts you want to remove.

Step 5 Click **Delete All Checked** to remove the user accounts. The system displays the names of the users you removed.

Managing groups

This section covers the following topics:

- [“Creating grid groups,”](#) below
- [“Adding users to groups”](#) on page 192
- [“Removing users from groups”](#) on page 194
- [“Refreshing imported groups”](#) on page 195
- [“Viewing and modifying group information”](#) on page 195
- [“Deleting groups”](#) on page 198

For information about the default groups in an Avaki domain, see the *Avaki Overture*.

Creating grid groups

To create a grid group, follow these steps.

Step 1 Log in as a member of the UserAdministrators group.

Step 2 Navigate to the Create Grid Group screen:

Home > User management > Create groups

The screenshot shows a dialog box titled "Create Grid Group". It has a light gray header bar with the title. Below the header, there are two input fields. The first is labeled "Group name:" and has a single-line text input box. The second is labeled "Description:" and has a multi-line text area with a vertical scrollbar on the right side. At the bottom of the dialog, there are two buttons: "Submit" and "Cancel", both with a light gray background and a slight shadow.

Step 3 Enter a name for the group and enter a description that explains the purpose of the group.

Step 4 Click **Submit**. The system displays a summary of the information you entered for the new group.

For instructions on specifying members for a group, see [“Adding users to groups,”](#) below.

Adding users to groups

To add users to a grid group, follow these steps. You can't add users to an imported group.

Note When an Avaki user is added to a group, the user must log out and log back in to enjoy any additional access enabled by the new group membership.

Step 1 Log in as a member of the UserAdministrators group.

Step 2 Navigate to the Select Group screen:

Home > User management > View and modify groups

Select Group

All groups

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

To modify a group's description, download a group interconnect ID, or add or remove group members, select the group and click the **View/Edit** button. To delete a group from a grid domain, select the group and click **Delete All Checked**. To configure the group ID (GID) that is mapped to a group account, select the group and click **Update GIDs** below.

Select	Group name	Authentication service	Authentication service type	View/Edit Group	Attributes	Security
<input type="checkbox"/>	Administrators	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	DataProviders	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	DatabaseAdministrators	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	DomainUsers	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	MessagingUsers	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	UserAdministrators	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	WaterBuffaloLodge	DefaultAuthService	Grid	View/Edit	Attributes	Security

Step 3 Click the **View/Edit** button for the group you want to add users to.

- Step 4** On the screen that appears, go to the Add Group Members area and click **Add Members**. The system displays the Add Grid User to Group screen.

Current Domain: Bedrock Current Server: ANTIMONYXP.sybase.com

Add users to group: WaterBuffaloLodge

Add users from grid domain: Bedrock ▾

Add Grid User to Group

All users

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

Add	Username	Authentication service	Authentication service type
<input type="checkbox"/>	Administrator	DefaultAuthService	Grid
<input type="checkbox"/>	MessagingUser	DefaultAuthService	Grid
<input type="checkbox"/>	barney	DefaultAuthService	Grid
<input type="checkbox"/>	betty	DefaultAuthService	Grid
<input type="checkbox"/>	fred	DefaultAuthService	Grid
<input type="checkbox"/>	wilma	DefaultAuthService	Grid

- Step 5** In the Add users from grid domain field, select the grid domain from which you want to add users.
- Step 6** Click the box in the Add column next to the name of each user that you want to add to the group.
- Step 7** Click **Submit**. The system displays the name of the group and a list of the users you've added.

Removing users from groups

To delete a user from a group in the grid, follow these steps. You cannot delete members from an imported group.

Step 1 Log in as a member of the UserAdministrators group.

Step 2 Navigate to the Select Group screen:

Home > User management > View and modify groups

Select Group

All groups
[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

To modify a group's description, download a group interconnect ID, or add or remove group members, select the group and click the **View/Edit** button. To delete a group from a grid domain, select the group and click **Delete All Checked**. To configure the group ID (GID) that is mapped to a group account, select the group and click **Update GIDs** below.

Select	Group name	Authentication service	Authentication service type	View/Edit Group	Attributes	Security
<input type="checkbox"/>	Administrators	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	DataProviders	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	DatabaseAdministrators	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	DomainUsers	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	MessagingUsers	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	UserAdministrators	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	WaterBuffaloLodge	DefaultAuthService	Grid	View/Edit	Attributes	Security

Step 3 Click the **View/Edit** button for the group you're interested in. The screen that appears includes a View and Remove Group Members area.

View and Remove Group Members

All users in group
[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

In the Select column, click the box beside the username and click **Submit** to remove a user from the group.

Select	Username	Domain	Authentication Service	Authentication Service Type
<input type="checkbox"/>	barney	Bedrock	DefaultAuthService	Grid
<input type="checkbox"/>	fred	Bedrock	DefaultAuthService	Grid

- Step 4** Click boxes in the Select column to choose the users you want to remove.
- Step 5** Click **Submit** to delete the selected users from the group. The system displays a list of the users you have deleted.

Refreshing imported groups

Groups that are imported into the grid from an external authentication service might need to be refreshed from time to time. Refreshing a group is an explicit operation you perform that updates it with any changes to its membership, bringing the group into sync with the corresponding group in the authentication service. When you refresh a group, you're essentially reimporting it from the authentication service.

Note Refreshing a group refreshes the accounts of all previously imported group members, and imports the accounts of any members who have been added since the group was last imported or refreshed.

To refresh an imported group, follow the instructions for importing. Refer to one of these procedures:

- [“Importing groups from an LDAP authentication service” on page 159](#)
- [“Importing groups from an NIS authentication service” on page 165](#)

Viewing and modifying group information

To display information about an Avaki group or to modify that information, follow these steps.

- Step 1** Log in as a member of the UserAdministrators group.
- Step 2** Navigate to the Select Group screen:

Home > User management > View and modify groups

Select Group						
All groups						
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z						
To modify a group's description, download a group interconnect ID, or add or remove group members, select the group and click the View/Edit button. To delete a group from a grid domain, select the group and click Delete All Checked . To configure the group ID (GID) that is mapped to a group account, select the group and click Update GIDs below.						
Select	Group name	Authentication service	Authentication service type	View/Edit Group	Attributes	Security
<input type="checkbox"/>	Administrators	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	DataProviders	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	DatabaseAdministrators	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	DomainUsers	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	MessagingUsers	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	UserAdministrators	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	WaterBuffaloLodge	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="button" value="Check All"/> <input type="button" value="Clear All"/>						
<input type="button" value="Delete All Checked"/> <input type="button" value="Update GIDs"/> <input type="button" value="Cancel"/>						

- Step 3** Click the **View/Edit** button for the group you're interested in. The system displays the Modify Groups screen.

Data Grid User Management

[Home](#) • [Manage Users & Groups](#) • Modify Groups

Current Domain: Bedrock **Current Server:** ANTIMONYXP.sybase.com

Modify Group Description

Group name: WaterBuffaloLodge
 GID for NFS:
 Description:

Download Group Interconnect ID

To send the group's interconnection ID to a remote domain, click **Download Interconnect ID**.

[Download Interconnect ID](#)

Add Group Members

To add a user to the group, click **Add Members**.

[Add Members](#)

View and Remove Group Members

All users in group

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

In the Select column, click the box beside the username and click **Submit** to remove a user from the group.

Select	Username	Domain	Authentication Service	Authentication Service Type
<input type="checkbox"/>	barney	Bedrock	DefaultAuthService	Grid
<input type="checkbox"/>	fred	Bedrock	DefaultAuthService	Grid

- Step 4** To edit the group's description, click in the text box for the Description field. To create a group descriptor you can use to give this group permissions in other domains, click Download Interconnect ID. To add users to the group, click **Add Members** and see

“Adding users to groups” on page 192. To remove users from the group, see “Removing users from groups” on page 194.

Deleting groups

To delete grid or imported groups, follow these steps.

Step 1 Log in as a member of the UserAdministrators group.

Step 2 Navigate to the Select Group screen:

Home > User management > View and modify groups

Select Group

All groups

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

To modify a group's description, download a group interconnect ID, or add or remove group members, select the group and click the **View/Edit** button. To delete a group from a grid domain, select the group and click **Delete All Checked**. To configure the group ID (GID) that is mapped to a group account, select the group and click **Update GIDs** below.

Select	Group name	Authentication service	Authentication service type	View/Edit Group	Attributes	Security
<input type="checkbox"/>	Administrators	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	DataProviders	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	DatabaseAdministrators	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	DomainUsers	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	MessagingUsers	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	UserAdministrators	DefaultAuthService	Grid	View/Edit	Attributes	Security
<input type="checkbox"/>	WaterBuffaloLodge	DefaultAuthService	Grid	View/Edit	Attributes	Security

Step 3 If you don't see the group you want to remove, click on a letter or on the **All groups** link to display more groups.

Step 4 Click boxes in the Select column to choose the group or groups you want to remove.

Step 5 Click **Delete All Checked** to remove the groups. The system displays the names of the groups you removed.

Basic tasks

This chapter tells you how to perform these tasks with Avaki software:

- “Logging in,” below
- “Accessing CIFS shares” on page 204
- “Naming Avaki objects” on page 207
- “Managing files and directories” on page 208
- “Using and managing categories” on page 221
- “Searching the data catalog” on page 231

Logging in

To sign in to an Avaki account, point your web browser to this address:

```
http://<machine>:7080
```

where <machine> is the name of the machine on which Avaki software is running.

7080 is the default port number. If a nondefault HTTP port has been configured, substitute that port number.

The Standard Sign-in screen appears.



[Help](#)

Welcome to the Avaki Data Grid

Existing Avaki users

To sign in, select the grid domain and authentication service, then enter your username and password.

To sign in to a grid domain that is connected to your domain but is not visible in the grid domain drop-down list, use the [Advanced Sign-in](#) screen.

Grid domain:

Authentication service:

Username:

Password:

[Advanced Sign-in Help](#)

To sign in to your local Avaki domain, follow the steps in “[Signing in to a local domain](#),” below. To sign in to an Avaki domain that is connected to your domain but is not visible in the Grid domain drop-down list, follow the steps in “[Signing in to a remote domain](#)” on page 202.

Signing in to a local domain

Follow these steps to sign in to a local Avaki domain:

- Step 1** Select your Avaki domain from the pull-down list.
- Step 2** Select the name of your authentication service from the pull-down list; DefaultAuthService is the default. If an LDAP-based or NIS authentication service has been integrated into your Avaki domain, select the name of the service.
- Step 3** Enter your username.
- Step 4** Enter your password.
- Step 5** Click **Sign In**.

The Welcome screen appears. You can return to this main menu from anywhere in the Avaki web interface by clicking the **Home** link at the top of each page.



Welcome, Administrator • [Logout](#) • [Help](#)

Welcome to the Avaki Data Grid

Current Domain: Bedrock **Current Server:** Antimony.avaki.local

[About my domain](#)



[Database provisioning](#)

Manage database connections, and make tables from and queries against connected databases visible in Avaki.



[Data integration](#)

Integrate data from disparate sources and manage the results.



[Category management](#)

Browse categories and manage object categorization.



[Data catalog management](#)

Manage files, directories, links, attributes, and security.



[User management](#)

Manage users and groups, integrate external authentication services.



[Server management](#)

Manage grid servers, share servers, proxy servers, and data grid access servers.



[Service management](#)

Manage monitoring, cache, search services, shares, and other services.



[Domain interconnection management](#)

Connect and disconnect grid domains.

Note If you are not logged in as a grid administrator, your Welcome screen menu contains fewer choices than are shown in this example.

Signing in to a remote domain

Follow these steps to sign in to an Avaki domain that is connected to your domain but is not visible in the grid domain drop-down list on the Standard Sign-in screen.

- Step 1** On the Standard Sign-in screen, click the Advanced Sign-in hyperlink. The Advanced Sign-in screen appears.



Existing Avaki users

To sign in to a grid domain that is connected to your domain, specify the grid domain, authentication service name and type, username, and password.

Grid domain:

Authentication service name:

Authentication service type: ▾

Username:

Password:

[Standard Sign-in Help](#)

- Step 2** Specify the Avaki domain to sign in to.
- Step 3** Specify the name of the authentication service to which your account in the target domain belongs.
- Step 4** From the pull-down list, select the authentication service's type: Grid (the default), NIS, or LDAP.
- Step 5** Enter your username.
- Step 6** Enter your password.
- Step 7** Click **Sign In**.

The Welcome screen appears. You can return to this main menu from anywhere in the Avaki web interface by clicking the **Home** link at the top of each page.



Welcome, Administrator • [Logout](#) • [Help](#)

Welcome to the Avaki Data Grid

Current Domain: Bedrock **Current Server:** Antimony.avaki.local

[About my domain](#)



[Database provisioning](#)

Manage database connections, and make tables from and queries against connected databases visible in Avaki.



[Data integration](#)

Integrate data from disparate sources and manage the results.



[Category management](#)

Browse categories and manage object categorization.



[Data catalog management](#)

Manage files, directories, links, attributes, and security.



[User management](#)

Manage users and groups, integrate external authentication services.



[Server management](#)

Manage grid servers, share servers, proxy servers, and data grid access servers.



[Service management](#)

Manage monitoring, cache, search services, shares, and other services.



[Domain interconnection management](#)

Connect and disconnect grid domains.

Note If you are not logged in as a grid administrator, your Welcome screen menu contains fewer choices than are shown in this example.

Accessing CIFS shares

A CIFS share is a directory or file that has been exported (shared) out of the data grid. A CIFS share can be mapped into a Windows file system as if it was a network drive. When you browse the Windows file system, CIFS shares look like—and can be accessed like—other files and directories.

Prerequisites

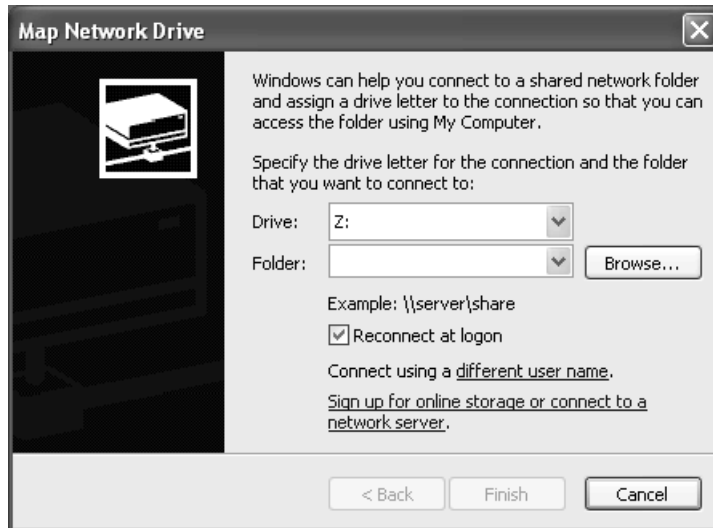
To enable you to access a CIFS share,

- You need a computer running Windows 2000 or Windows XP that has network connectivity to the Avaki data grid access server (DGAS).
- An Avaki administrator or data provider must set up the CIFS share through a DGAS.
- You need to know the name or IP address of the DGAS machine and the name of the CIFS share.
- You must have a user account in the Avaki domain that the DGAS belongs to.
- An Avaki administrator or data provider must add your grid user account to the ACL (access control list) for the CIFS share's directory in the data catalog.
- If your grid user account is imported into the grid from an NIS directory service, you must log in to Avaki (either the web UI or the command line) once before you can connect to a CIFS share. Every time you change your password, you must log in to Avaki again to refresh your credentials to map or reconnect to CIFS shares.
- An Avaki administrator can set an admission policy that allows your Windows machine (or all machines on its network) to access files through the DGAS. If your grid account is imported from an LDAP service, the Avaki administrator *must* set such an admission policy.

Mapping a CIFS share

Follow these steps to map a CIFS share to a network drive:

- Step 1** Log in to the Windows machine on which you want to access the CIFS share.
- Step 2** Open the Map Network Drive tool.



- Step 3** In the Drive field, use the pull-down to choose an unused drive letter.
- Step 4** In the Folder field, enter a path of this form:

```
\\<DGAS-name>\<CIFS-share-name>
```

where

<DGAS-name> is the name of the DGAS machine that's serving the CIFS share

<CIFS-share-name> is the name of the share. Include the path to a subdirectory of the share if you wish to connect directly to the subdirectory. The default CIFS share is SHARES, which is a link to the grid's /Shares directory. If the DGAS is called DGAS1, you might enter the following in the Folder field:

```
\\DGAS1\SHARES
```

- Step 5** If you don't want to remount this CIFS share whenever you log in to this machine, click to uncheck the Reconnect at logon box.

Note If your account is imported from an LDAP directory service, you must access CIFS shares from a machine that is allowed access in the DGAS admission policies.

Step 6 Click **Connect** using a **different user name**. The **Connect As...** dialog box appears.



Step 7 In the **User name** field, enter your grid account name.

Note If your Avaki administrator has not set admission policies to allow your Windows machine to access files through the DGAS, *and* if your account was created in the grid or imported from an NIS directory service, use this syntax to enter your user name:

```
<auth-service-name>@<auth-service-type>\<grid-user-name>
```

where

<auth-service-name> is the name of the grid authentication service to which your grid user account belongs—for example, DefaultAuthService

<auth-service-type> is Grid or Nis

<grid-user-name> is your grid user account name

Step 8 In the **Password** field, enter the password for your grid user account.

Step 9 Click **OK**.

Step 10 In the **Map Network Drive** window, click **Finish** to map the CIFS share.

Naming Avaki objects

Case sensitivity

Avaki is case sensitive with respect to the names of objects in the data catalog, including grid directories, links, Avaki shares, files and directories in Avaki shares, CIFS shares, categories, servers, user accounts, groups, database operations, data services, SQL views, view generators, and generated views. However, when you use Avaki to share files from an operating system (such as Windows) that is not case sensitive, you remain bound by OS's capabilities—for example, you can't create files called Cat and cat in the same directory of a Windows-based Avaki share.

Character restrictions

The names of files and directories in Avaki shares are governed by restrictions imposed by the operating system.

For the names of other Avaki objects, including grid directories, links, Avaki shares, CIFS shares, categories, servers, user accounts, groups, database operations, data services, SQL views, view generators, and generated views, there are restrictions on the characters you can use:

- You can use letters, digits, and underscores (`_`).
- Do not use spaces, dots (`.`), hyphens (`-`), or other special characters.

Managing files and directories

This section covers the following topics:

- [“Creating grid directories,”](#) below
- [“Moving files and directories”](#) on page 210
- [“Renaming files and directories”](#) on page 212
- [“Copying files and directories”](#) on page 214
- [“Making a local copy of a grid file”](#) on page 216
- [“Linking files and directories”](#) on page 217
- [“Deleting files and directories”](#) on page 220

Note For instructions on setting up, managing, and disabling Avaki shares, see [Chapter 8, “Managing Avaki shares”](#).

Creating grid directories

Follow these steps to create a directory in an Avaki domain.

Caution When you create a directory inside an Avaki share, you create the new directory in the source file system—not in the data catalog only.

Step 1 Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

Directory names

<input type="checkbox"/>	/	Attributes	Security	
<input type="checkbox"/>	GeneratedViews	Attributes	Security	Categories
<input type="checkbox"/>	Interconnects	Attributes	Security	Categories
<input type="checkbox"/>	Metadata	Attributes	Security	Categories
<input type="checkbox"/>	Shares	Attributes	Security	Categories
<input type="checkbox"/>	System	Attributes	Security	Categories
<input type="checkbox"/>	WSDLs	Attributes	Security	Categories
<input type="checkbox"/>	Categories	Attributes	Security	Categories

- Step 2** On the Browse Directories screen, click on directory names (which are flanked by folder icons) to navigate to the parent directory—the directory in which you want your new directory to appear. The new directory will be created in the parent directory that's displayed at the top of the Browse Directories area.
- Step 3** Click **Create Directory**. the Create New Directory screen appears.

Create New Directory

Parent directory: /System/Domains/CherylDomain

New directory name:

- Step 4** In the New directory name box, enter a name for the directory you're creating.
- Step 5** Click **Submit**. The system displays the Browse Directories screen again, showing your newly created directory.

By default, the permissions on a new Avaki directory are set so that the user who created the directory (who is the owner) can read, write, execute, and delete it and its con-

tents; members of the DomainUsers group who are in the current Avaki domain can read and execute the directory and its contents, but they cannot write or delete. Users in other Avaki domains have no permissions.

For instructions on setting permissions on Avaki directories, see [“Using access control lists” on page 237](#).

Moving files and directories

Follow these steps to move one or more files or Avaki directories to a new location in the data catalog.

Note You cannot move a file or directory into an Avaki share or out of an Avaki share. (You can move objects within an Avaki share.) To put a directory or file into an Avaki share, you must copy it. See [“Copying files and directories” on page 214](#). To remove a directory or file from an Avaki share, you must delete it. See [“Deleting files and directories” on page 220](#).









Step 1 Navigate to the Browse Directories screen:

Home > Data catalog management

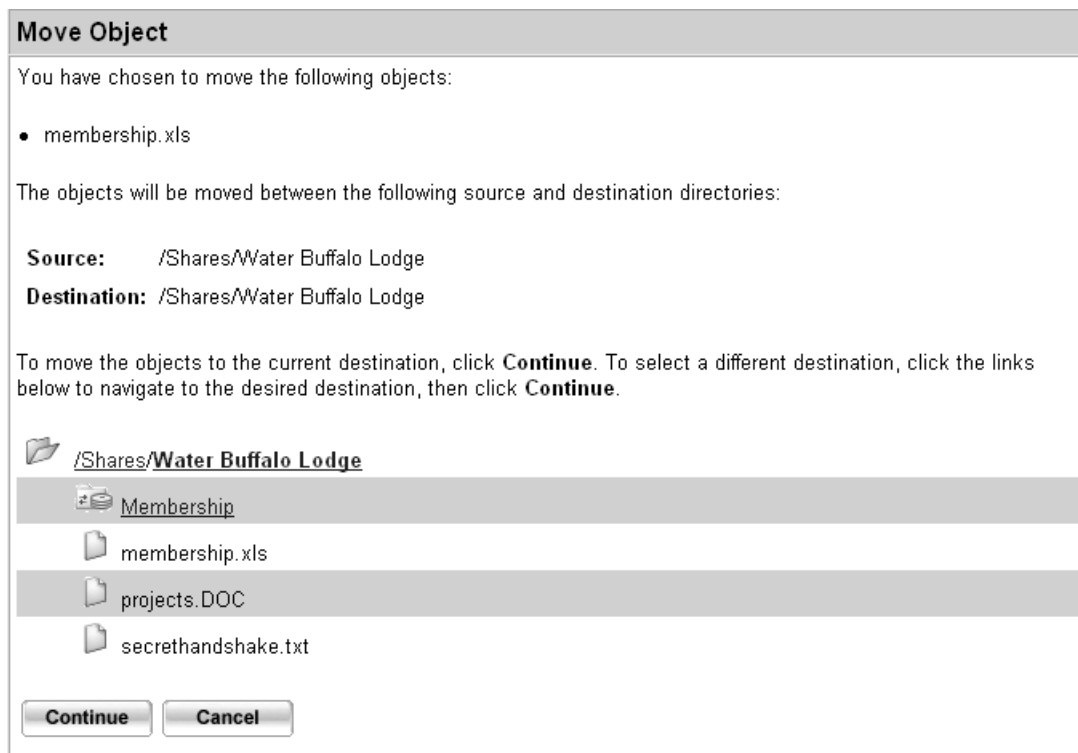
Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>		/	Attributes	Security	
<input type="checkbox"/>		GeneratedViews	Attributes	Security	Categories
<input type="checkbox"/>		Interconnects	Attributes	Security	Categories
<input type="checkbox"/>		Metadata	Attributes	Security	Categories
<input type="checkbox"/>		Shares	Attributes	Security	Categories
<input type="checkbox"/>		System	Attributes	Security	Categories
<input type="checkbox"/>		WSDLs	Attributes	Security	Categories
<input type="checkbox"/>		Categories	Attributes	Security	Categories

- Step 2** On the Browse Directories screen, click on directory names to navigate to the file, directory, or other Avaki object that you want to move.
- Step 3** Click the check box to the left of the target object's name. You can select more than one object if you wish.
- Step 4** Click **Move All Checked**. The Move Object screen appears.



- Step 5** To move the target object(s) to another Avaki directory within the same Avaki share, click on directory names to navigate to the desired location—for example, click the folder icon to move up to the root Avaki directory—then click **Continue**.

- Step 6** By default, moved objects retain their original names when moved. If you want to rename an object, specify the new name in the New Name box, then click **Move**.

Move Object

You are ready to move the selected objects between the following source and destination directories:

Source: /Shares/Water Buffalo Lodge

Destination: /Shares/Water Buffalo Lodge/Membership

By default, the objects will retain their original names when they are moved. To rename an object, specify the new name below. Click **Move** to place the objects in the destination directory.

Original Name	New Name
membership.xls	membership.xls

The system displays the Browse Directories screen with the moved objects in their new location.

Renaming files and directories

Follow these steps to change the name of a grid directory or a file.

Caution When you rename a file or directory within an Avaki share, the change is reflected in the source file system—not in the data catalog only.

Note You cannot rename an Avaki share, but you can rename files and directories within the share.

- Step 1** Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>	/	Attributes	Security	
<input type="checkbox"/>	GeneratedViews	Attributes	Security	Categories
<input type="checkbox"/>	Interconnects	Attributes	Security	Categories
<input type="checkbox"/>	Metadata	Attributes	Security	Categories
<input type="checkbox"/>	Shares	Attributes	Security	Categories
<input type="checkbox"/>	System	Attributes	Security	Categories
<input type="checkbox"/>	WSDLs	Attributes	Security	Categories
<input type="checkbox"/>	Categories	Attributes	Security	Categories

- Step 2** On the Browse Directories screen, click on directory names to navigate to the Avaki directory containing the files, directories, or other Avaki objects that you want to rename.
- Step 3** Click the check box to the left of the target file(s) or directory name(s).
- Step 4** Click **Rename All Checked**. The Rename Object screen appears.

Rename Object

Please enter the new name for each of the items you have chosen to rename.

Current Name	New Name
projects.DOC	<input type="text" value="projects.DOC"/>
xml-ouput.dtd	<input type="text" value="xml-ouput.dtd"/>

- Step 5** Enter a new name in the New Name field for each object you're renaming.
- Step 6** Click **Continue** to save the new name(s). The system displays the Browse Directories screen showing the new names.

Copying files and directories

Follow these steps to copy one or more files, Avaki directories, or Avaki shares to a new location in the data catalog. (For instructions on copying a data grid file to your local file system, see [“Making a local copy of a grid file”](#) on page 216.)

Caution When you copy a file or directory into an Avaki share, out of an Avaki share, or within an Avaki share, the change is reflected in the source file system—not in the data catalog only. You cannot copy an entire shared directory.

Note You cannot copy files into unshared grid directories; files can exist only within Avaki shares.

Step 1 Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

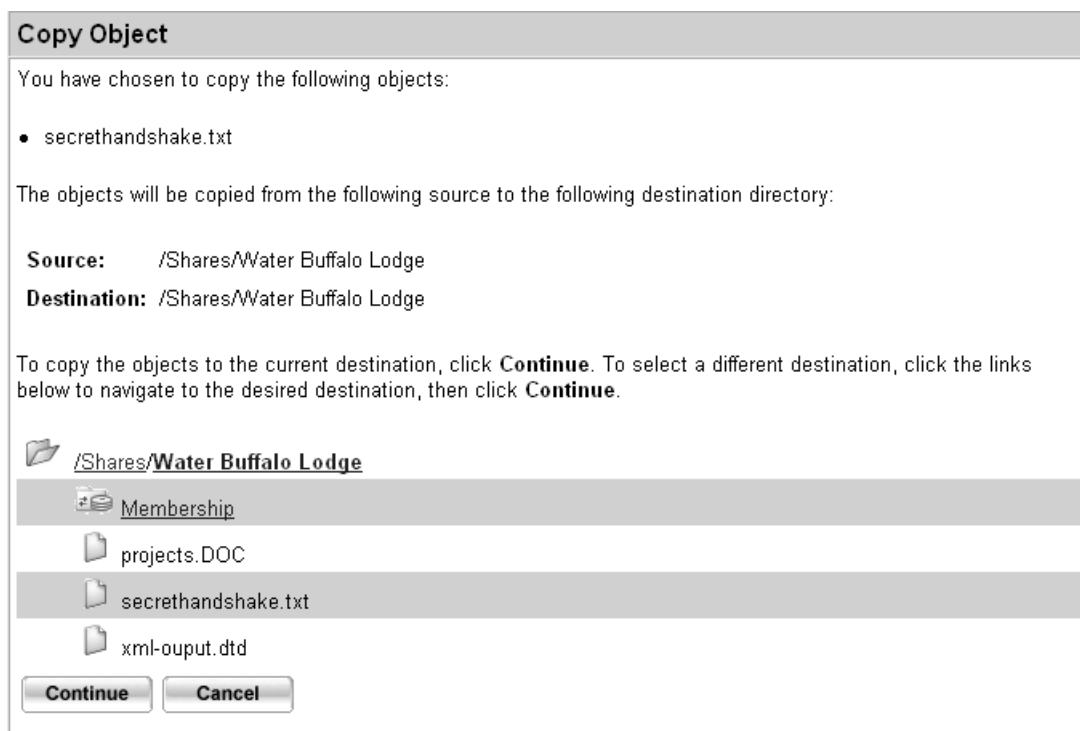
You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>	/	Attributes	Security
<input type="checkbox"/>	GeneratedViews	Attributes	Security Categories
<input type="checkbox"/>	Interconnects	Attributes	Security Categories
<input type="checkbox"/>	Metadata	Attributes	Security Categories
<input type="checkbox"/>	Shares	Attributes	Security Categories
<input type="checkbox"/>	System	Attributes	Security Categories
<input type="checkbox"/>	WSDLs	Attributes	Security Categories
<input type="checkbox"/>	Categories	Attributes	Security Categories

Step 2 On the Browse Directories screen, click on directory names to navigate to the Avaki directory containing the files, directories, or other Avaki objects that you want to copy.

Step 3 Click check boxes on the left to select the desired objects.

Step 4 Click **Copy All Checked**. The Copy Object screen appears.



- Step 5** If you want to copy the selected object(s) to the same directory, click **Continue**. If you want to copy the object(s) to another location, click on directory names to navigate to the desired destination (for example, click **Shares**), then click **Continue**.
- Step 6** By default, copied objects have the same names as the originals. If you want to rename an object, specify the new name in the Name of Copy box, then click **Copy**. The system displays the Browse Directories screen showing the copied objects in their new locations.

Making a local copy of a grid file

Follow these steps to save a local copy of a file that's shared into the data catalog.









Step 1 Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>	 /		Attributes	Security
<input type="checkbox"/>	 GeneratedViews		Attributes	Security Categories
<input type="checkbox"/>	 Interconnects		Attributes	Security Categories
<input type="checkbox"/>	 Metadata		Attributes	Security Categories
<input type="checkbox"/>	 Shares		Attributes	Security Categories
<input type="checkbox"/>	 System		Attributes	Security Categories
<input type="checkbox"/>	 WSDLs		Attributes	Security Categories
<input type="checkbox"/>	 Categories		Attributes	Security Categories

Step 2 On the Browse Directories screen, click on directory names to navigate to the file that you want to copy.

Step 3 Right-click on the target file name. From the pop-up menu that appears, select **Save Target As...** (in Internet Explorer) or **Save Link As...** (in Netscape or Konqueror) or **Save Link Target As...** (in Mozilla). The web browser opens a dialog box in which you can save the file in the local file system.

Linking files and directories

You can create two types of links in the data catalog:

- **Hard links**
A hard link is an alternate name for a file or directory. If a hard link points to a file location in the current domain and you move or change the file's original name, the hard link will still know where to find it. However, if the hard link points to a location in a remote domain and a file or directory at that location is moved, deleted, or renamed, the hard link leads nowhere.
- **Soft links**
A soft link is a pointer to a particular location (name) in the data catalog. If the file or directory at that location is moved, deleted, or renamed, the soft link leads nowhere. Avaki searches do not follow soft links.

Note Neither hard nor soft links created in the data catalog are created in the source file system.

Follow these steps to create hard or soft links from one or more files, directories, or other Avaki objects to another location (name) in the data catalog.









Step 1 Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

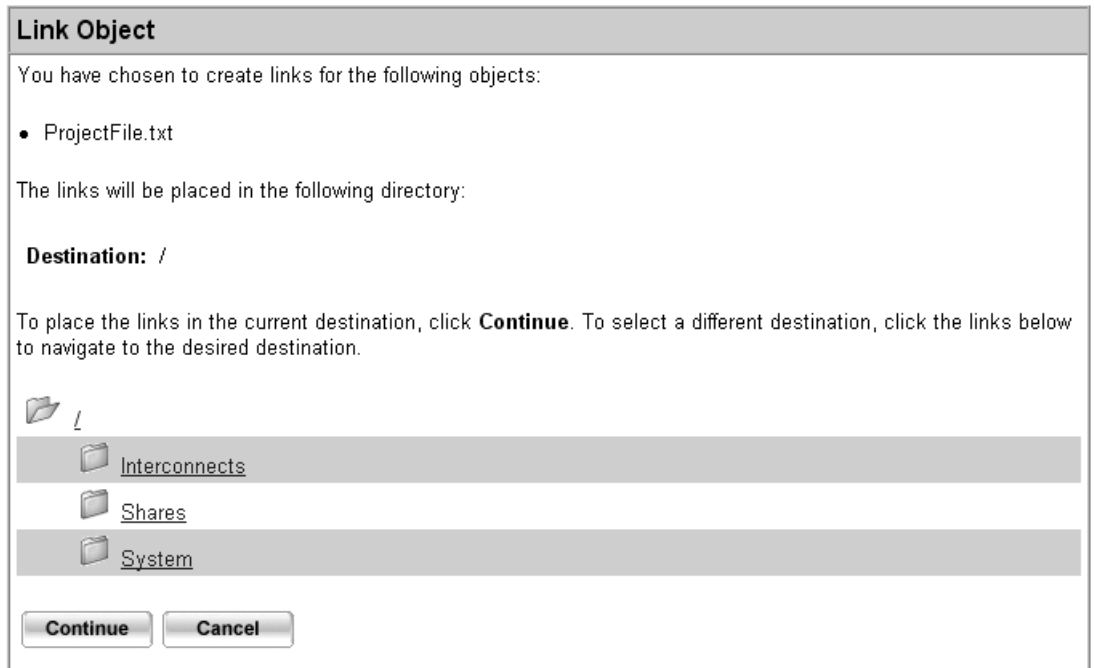
The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>	 /	Attributes	Security	
<input type="checkbox"/>	 GeneratedViews	Attributes	Security	Categories
<input type="checkbox"/>	 Interconnects	Attributes	Security	Categories
<input type="checkbox"/>	 Metadata	Attributes	Security	Categories
<input type="checkbox"/>	 Shares	Attributes	Security	Categories
<input type="checkbox"/>	 System	Attributes	Security	Categories
<input type="checkbox"/>	 WSDLs	Attributes	Security	Categories
<input type="checkbox"/>	 Categories	Attributes	Security	Categories

- Step 2** On the Browse Directories screen, click on directory names to navigate to the Avaki objects that you want to select as the destination for your link.
- Step 3** Click check boxes on the left to select the objects you want to link.

Step 4 Click **Link All Checked**. The Link Object screen appears.



Step 5 If you want to link the selected objects into / (the root directory of the data catalog), click **Continue**. If you want to link the objects into another location, click on directory names to navigate to the desired location, then click **Continue**.

Note To remove either a hard link or a soft link, delete it as you would delete a file. See [“Deleting files and directories” on page 220](#).

Step 6 By default, linked objects have the same name as the originals. If you want to rename an object, specify the new name in the New Link Name box, and then click **Make Hard Link** or **Make Soft Link**. The system displays the Browse Directories screen showing the new link(s).

Deleting files and directories

Caution When you delete a shared file or directory, the change is reflected in the source file system—not in the data catalog only. If you want to shut down an Avaki share without affecting the source files, refer to [“Disconnecting Avaki shares” on page 288](#) for instructions.

Note You cannot delete the parent directory of an Avaki share.

Follow these steps to delete Avaki objects such as files, Avaki directories, links, or the contents of Avaki shares.

Step 1 Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

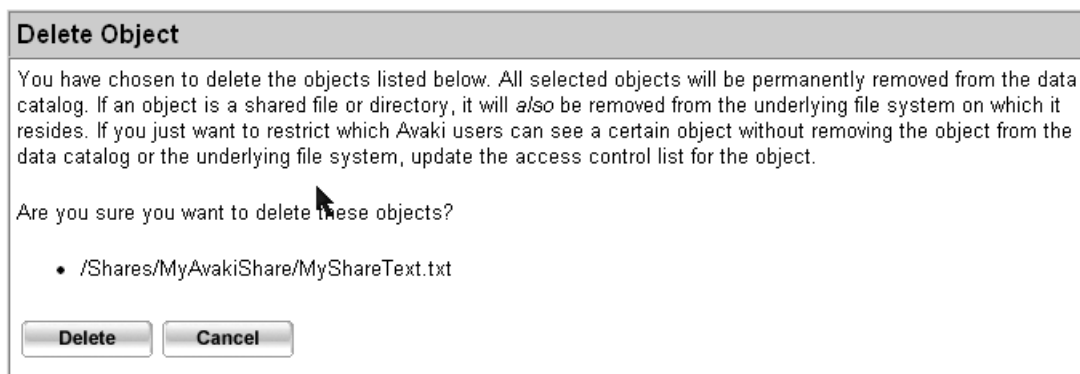
You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>	/	Attributes	Security
<input type="checkbox"/>	GeneratedViews	Attributes	Security Categories
<input type="checkbox"/>	Interconnects	Attributes	Security Categories
<input type="checkbox"/>	Metadata	Attributes	Security Categories
<input type="checkbox"/>	Shares	Attributes	Security Categories
<input type="checkbox"/>	System	Attributes	Security Categories
<input type="checkbox"/>	WSDLs	Attributes	Security Categories
<input type="checkbox"/>	Categories	Attributes	Security Categories

Step 2 On the Browse Directories screen, click on directory names to navigate to the objects that you want to delete.

Step 3 Click check boxes to the left to select the objects.

Step 4 Click **Delete All Checked**. The Delete Object screen appears.



Step 5 Click **Delete** to confirm that you want to remove the object(s). The system displays the Browse Directories screen without the deleted object(s).

Using and managing categories

Categories let you classify and organize the contents of your data catalog.

In this section:

- [“About categories,”](#) below
- [“Browsing categories”](#) on page 222
- [“Creating categories”](#) on page 224
- [“Adding grid objects to a category”](#) on page 226
- [“Removing grid objects from a category”](#) on page 228
- [“Deleting categories”](#) on page 230

About categories

Categories are a lot like Avaki directories: they serve as containers for the objects in your data catalog. Anything in the data catalog—views, data services, shared files, even Avaki directories themselves—can be assigned to a category. Categories are hierarchical, they have attributes, and Avaki ACLs regulate access to them. You can think of categories as an alternate, parallel set of directories.

The difference between categories and grid directories is that categories exist entirely for your convenience. While every Avaki object is represented in a directory in the data catalog, assigning objects to categories is optional. When an Avaki object is added to a category, you can navigate to it through either the category browser or the Avaki directory browser.

Categories make it easier for Avaki users to find what they need. Here are some ways to use them:

- Objects that are buried many layers down in the Avaki directory hierarchy, or that reside in large Avaki directories where individual entries are hard to find, can be placed in one or more categories that are tailored to the needs of the people who use them.
- You can use categories to make selected Avaki objects accessible to users who don't have the permissions to reach the objects through the data catalog. (To access an Avaki object, a user needs permissions not only on the object itself, but on all parent objects in the path to the target object. For example, to read the file `Dinosaurs.doc` in the Avaki directory `/Shares/pets`, you must have read permission on the root directory `/`, `/Shares`, `/Shares/pets`, and `/Shares/pets/Dinosaurs.doc`. This rule applies to categories as well as Avaki directories.)
- Categories give you great flexibility. When you rearrange categories and their contents, it has no effect on the underlying Avaki objects or directories.

Some objects are automatically assigned to categories when you create the objects:

- Data services are assigned to the category `/ViewLibrary/DataServices`.
- Database connectors are assigned to the category `/ViewLibrary/DatabaseServices`.
- Database operations are assigned to the category `/ViewLibrary/DatabaseServices/<database-connector-name>`.

We consider `ViewLibrary` and the root category to be default categories.

Browsing categories

By default, users who are not members of the `DataProviders` group have only read permission on the default categories (the root category and `ViewLibrary`)—that is, users can list the contents of the default categories and read any objects for which permissions have not been explicitly denied.

By default, the owner and members of the DataProviders group have read, write, execute, and delete permission on new categories. All other users have read permission on new categories.

In the web UI, you can look at Avaki categories in several ways, provided you have the appropriate permissions:

From the category browser, you can list categories and their contents, add Avaki objects to categories and remove them, and delete categories.

To open the category browser and display the hierarchy of categories, log in to the web UI and select **Category management** from the Welcome screen.

(You can also reach the category browser through the directory browser. Click **Categories**, the last entry on the top-level Browse Directories screen (Home > Data catalog management > Categories).

From the directory browser, if you want to display a list of the categories to which an Avaki object has been assigned, to add the object to a category, or remove it from a category, you can use the directory browser. Select Home > Data catalog management, then navigate to the Avaki object you're interested in and click its **Categories** link.

Note In a newly installed Avaki domain, there are only two categories, the root category and the ViewLibrary category. Until you (or an Avaki administrator or data provider) have added and populated categories, there won't be much to see in the category browser.

Browse Categories

Grid domain:

Current Category: Category Root

Description: Root category for domain Bedrock

Subcategories:

ViewLibrary	no description	Attributes	Security	Delete
-----------------------------	----------------	----------------------------	--------------------------	------------------------

The category tree is populated in two ways:

- You can add categories as described in [“Creating categories” on page 224](#), and populate the categories as described in [“Adding grid objects to a category” on page 226](#).

- You can set up database operations or data services. When database operations and data services are added to the grid domain, they are linked into the category tree under ViewLibrary.

The category browser and the Avaki directory browser are linked because categories can contain Avaki directories. If you browse to a category that contains a directory, clicking on the directory takes you to that directory in the directory browser.

Note, however, that categories and directories are fundamentally different. Though the two hierarchies may seem to flow into one another when you add directories to categories, they remain distinct. You cannot create directories in categories or vice versa.

Clicking on any object's **Categories** link, in either the category browser or the catalog browser, takes you to the View Categories page, below. This page displays a list of the categories to which the object belongs, lets you delete the object from any of its current categories, and lets you add it to other categories (click **Browse** to select a category). Each category in the list is linked, and clicking the link takes you to the category browser proper.

View Categories

/Shares/Water Buffaloes/xmloutput.dtd belongs to the following categories:

Name	Description	Remove
Bedrock.net	no description	Remove

Add this object to a category:

Creating categories

In a newly installed data catalog, only members of the DataProviders group can create new categories.

If no categories have been created yet, consider setting up some basic categories in the root category—for projects, users, or departments, for example—so that ACLs can be configured to regulate access appropriately.

Follow these steps to create a category.

Step 1 Log in to Avaki as a user who has write permission in the parent category of the category you want to create.

Step 2 Navigate to the category browser:

Home > Category management

Note You can also reach category tools through the Avaki directory browser. Click Home > Data catalog management, then navigate to the Avaki object you're interested in and click its **Categories** link.

Step 3 To navigate to the parent category, click on a category name in the Subcategories field.

Step 4 When you reach the category where you want to create your new category, click **Create Subcategory**. The system displays the Create New Category screen.

Step 5 Enter a name for the category you're creating.

Step 6 (Optional.) Enter a description for the new category.

- Step 7** Click **Submit** to create the category. The system displays the category browser with your new category.

Browse Categories

Grid domain: Bedrock ▼

Current Category: Category Root
Description: Root category for domain Bedrock

Subcategories:

ViewLibrary	no description	Attributes	Security	Delete
Water Buffalo Lodge	no description	Attributes	Security	Delete
Slate Quarry Co.	no description	Attributes	Security	Delete

- Step 8** To change the ACLs, click the **Security** link for the new category. See [“Using access control lists” on page 237](#) for information on how ACLs work and how to change them.

Adding grid objects to a category

By default, only the owner or a member of the DataProviders group can add grid objects to an existing category. If you're not an owner or DataProvider, an administrator must add you to the access control lists (ACLs) of any categories where you want to add or remove content.

Follow these steps to assign an Avaki object (a view, data service, database operation, file, or directory, for example) to a category.

- Step 1** Log in to Avaki as a user who has write permission in the category to which you want to add objects.

Step 2 Navigate to the category browser:

Home > Category management

Browse Categories

Grid domain: Bedrock

Current Category: Category Root

Description: Root category for domain Bedrock

Subcategories:

ViewLibrary	no description	Attributes	Security	Delete
Water Buffalo Lodge	no description	Attributes	Security	Delete
Slate Quarry Co.	no description	Attributes	Security	Delete

Note You can also reach category tools through the Avaki directory browser. Click Home > Data catalog management, then navigate to the Avaki object you're interested in and click its **Categories** link.

Step 3 To navigate the category hierarchy, click on a category name in the Subcategories field (Water Buffalo Lodge, for example). The browser displays information about the current category and a list of its contents, if any.

Browse Categories

Grid domain: Bedrock

Current Category: /Water Buffalo Lodge [View parent category](#)

Description: no description

Subcategories:

Barney	no description	Attributes	Security	Delete
Fred	no description	Attributes	Security	Delete

Objects in Current Category:

<input type="checkbox"/>	project.doc	Attributes	Security	Categories
--------------------------	-----------------------------	----------------------------	--------------------------	----------------------------

- Step 4** When you reach your target category, click **Add to Category** at the bottom. The Add Object to Category screen appears.

Add Object to Category

Enter or **Browse** to the file or other object you wish to add to category **/Water Buffalo Lodge**.

Object:

- Step 5** Browse to and select the object you want to add, or enter its data catalog path (for example, /Shares/Wilma/myfile) in the Object field.
- Step 6** Click **Submit**. The category browser displays the current category with the new object added.
- Step 7** To change the ACLs, click the **Security** link for the new object. See [“Using access control lists” on page 237](#) for information on how ACLs work and how to change them.

Removing grid objects from a category

Follow these steps to take an Avaki object out of a category.

Note Removing an object from a category does not delete the object from the data catalog.

- Step 1** Log in to Avaki as a member of the Administrators group or as a user who has write permission for the category from which you want to remove objects.

Step 2 Navigate to the category browser:

Home > Category management

Browse Categories

Grid domain: Bedrock

Current Category: Category Root

Description: Root category for domain Bedrock

Subcategories:

ViewLibrary	no description	Attributes	Security	Delete
Water Buffalo Lodge	no description	Attributes	Security	Delete
Slate Quarry Co.	no description	Attributes	Security	Delete

Note You can also reach category tools through the Avaki directory browser. Click Home > Data catalog management, then navigate to the Avaki object you're interested in and click its **Categories** link.

Step 3 To navigate the category hierarchy, click on a category name in the Subcategories field (Water Buffalo Lodge, for example). The browser displays information about the current category and a list of its contents, if any.

Browse Categories

Grid domain: Bedrock

Current Category: [Water Buffalo Lodge](#) [View parent category](#)

Description: no description

Subcategories:

Barney	no description	Attributes	Security	Delete
Fred	no description	Attributes	Security	Delete

Objects in Current Category:

<input type="checkbox"/>	project.doc	Attributes	Security	Categories
--------------------------	-----------------------------	----------------------------	--------------------------	----------------------------

Step 4 When you reach your target category, click the box to the left of the object you want to remove.

- Step 5** Click **Remove All Checked from Category** at the bottom. The category browser displays the current category with the object removed.

Deleting categories

Follow these steps to delete a category.

Note It's not necessary to remove a category's contents before deleting the category.

- Step 1** Log in to Avaki as a member of the Administrators group or as a user who has delete permission for the category you want to remove.

- Step 2** Navigate to the category browser:

Home > Category management

Browse Categories

Grid domain:

Current Category: Category Root

Description: Root category for domain Bedrock

Subcategories:

ViewLibrary	no description	Attributes	Security	Delete
Water Buffalo Lodge	no description	Attributes	Security	Delete
Slate Quarry Co.	no description	Attributes	Security	Delete

Note You can also reach category tools through the Avaki directory browser. Click Home > Data catalog management, then navigate to the Avaki object you're interested in and click its **Categories** link.

- Step 3** To navigate the category hierarchy, click on a category name in the Subcategories field (Water Buffalo Lodge, for example). The browser displays information about the current category and a list of its contents, if any.
- Step 4** To get rid of a category, click its **Delete** link. The category browser displays the parent category with the deleted category removed.

Searching the data catalog

You can search the data catalog for information stored in the attributes of Avaki objects. For example, you can perform a search for all files owned by a particular user, or files last changed on a certain date, or files with a given string attribute.

This section covers these topics:

- [“Creating a search service,”](#) below
- [“Performing a search” on page 233](#)

For more about attributes, see [“Using attributes” on page 246](#).

Creating a search service

A search service maintains a collection or index of all the attributes for all the Avaki objects (views, data services, Avaki shares, files, and so forth) in a specified portion of the data catalog. It is impossible to search an Avaki directory or a share that is not covered by a search service.

To create a search service for an Avaki directory or an Avaki share, you must:

- Have read permission on the object, and
- Be a member of either the DataProviders group or the Administrators group

Follow these steps to create a search service:

Step 1 Navigate to the Create Search Service screen:

Home > Service management > Create search service

Create Search Service

A search service collects attributes for all objects in a particular directory. To create a search service, specify a name for the service, the root directory from which the attributes will be collected, the reindex interval (indicating how often the attribute information will be updated), the node on which the service will be created, and a description of the search service. Note that a reindex interval of 0, which is the default, means that attribute information will never be updated after it is first collected.

Search service name:

Root directory: [Browse](#)

Reindex interval (in seconds):

Server: ▼

Description (optional):

Step 2 Fill in the form:

- Search service name: Specify a name for the search service.
- Root directory: Click **Browse** and navigate to the directory whose contents the search service will index, then click **Continue**.
- Reindex interval: The amount of time in seconds between reindex operations, in which the search service rereads all the attributes and refreshes its search index. The default value is 0. Set the reindex interval to a low value if your data catalog contains a large number of objects or if attribute values change frequently. Use a higher value if your data catalog contains a relatively small number of objects.

Note When you determine the reindex interval, consider how long it takes to reindex, how many CPU cycles are required, and how up-to-date your information must be. For a large directory with thousands of files, reindexing can take a significant amount of time. Consider using a dedicated machine for the grid server on which a large directory's search service resides.

- Server: From the pull-down menu, select the name or IP address of the grid server on which this search service's root directory resides.

- Description (optional): Enter some information about what this search service covers. This might be files belonging to a particular user or group, or those associated with a project.

Step 3 Click **Create Service** to create the search service. A confirmation screen appears when the search service is complete.

Performing a search

Before you can search, a search service must be created for the portion of the data catalog to be searched. For details, see [“Creating a search service” on page 231](#).

Note Searches do not traverse soft links in grid directories.

Follow these steps to perform a search in the data catalog.

Step 1 Navigate to the View/Use Search Service screen:

Home > Service management > View search services

View/Use Search Service

The following search services are available on your grid domain. You may view more information about a search service, execute a search, or delete a search service.

Search Service Name	View Details	Execute Search	Delete Service	Attributes	Security
MySearchService	View	Search	Delete	Attributes	Security

- Step 2** In the Execute Search column, click **Search** for the search service you want to use. The Specify Search Options screen appears.

Specify Search Options

Search service: mySearchService

You may search for a single attribute or for several attributes. To search for multiple attributes, specify the criteria for each attribute, then select AND or OR. If you select AND, the results will include attributes that meet the criteria specified for all the attributes in the query. If you select OR, the results will include attributes that meet the criteria specified for any attribute in the query.

Logical Operator	Description
=	is equal to
<>	is not equal to
>	is greater than
>=	is greater than or equal to
<	is less than
<=	is less than or equal to

Attribute:	Type:	Logical operator:	Value (optional):	
<input type="text"/>	String ▾	= ▾	<input type="text"/>	<input checked="" type="radio"/> AND <input type="radio"/> OR
<input type="text"/>	String ▾	= ▾	<input type="text"/>	
<input type="text"/>	String ▾	= ▾	<input type="text"/>	

- Step 3** Fill in the form. Every field but Value is required:

- **Attribute:** Enter the name of the attribute you want to search on. For example, to search for a file by name, enter system/name; or to search on the file's owner, enter system/Owner. (For instructions on listing attributes, see [“Viewing attributes” on page 247.](#))
- **Type:** Choose the type of the attribute from the pull-down menu.

- Logical operator: Choose one of the following operators from the pull-down menu:

Logical operator	Description
=	is equal to
<>	is not equal to
>	is greater than
>=	is greater than or equal to
<	is less than
<=	is less than or equal to

- Value: Enter the value of the attribute. You can use a wildcard to indicate a character or set of characters to use in searches. A wildcard can indicate a point in a search string where any character or string is a good match, where any of one or more specified characters is a good match, or where any of the specified characters is not an acceptable match. This table shows how to use wildcards in searches:

Wildcard	Description
% or *	Match any character or string at this position in the search pattern
_ or ?	Match any single character at this position in the search pattern

For example, you could use either the % or the * wildcard to search for an object's owner. The following query finds all system/Owner attributes that have a value of Administrators:


Attribute:	Type:	Logical operator:	Value (optional):
system/Owner	String	=	%Administrators%

- Boolean operator: Select AND to find Avaki objects whose attributes meet all the criteria you specify on this page. Select OR to find objects whose attributes meet any of the criteria you specify.

Step 4 Click **Search** to execute the search. The Search Results screen appears, with a list of the objects that match your search criteria.

Search Results

The following objects match the search criteria. Click on an object's path to view the Browse Directories page for the object (if it is the object a directory), or download the object (if the object is a file).

Object Path	View Attributes
 /System/Domains/CherylDomain/myProjects	View

Deleting a search service

Follow these steps to delete a search service:

Step 1 Navigate to the View/Use Search Service screen:

Home > Service management > View search services

View/Use Search Service

The following search services are available on your grid domain. You may view more information about a search service, execute a search, or delete a search service.

Search Service Name	View Details	Execute Search	Delete Service	Attributes	Security
MySearchService	View	Search	Delete	Attributes	Security

Step 2 In the Delete Service column, click **Delete** for the search service you want to delete.

Step 3 In the confirmation dialog box that appears, click **OK**. The search service is deleted.

Using ACLs and attributes

This chapter tells you how to perform these tasks with Avaki software:

- [“Using access control lists,”](#) below
- [“Using attributes”](#) on page 246

Using access control lists

This section covers the following topics:

- [“Viewing ACLs”](#) on page 238
- [“Modifying permissions in ACLs”](#) on page 240
- [“Adding users and groups to ACLs”](#) on page 243

For a discussion of how permissions and ACLs work in Avaki, see the *Sybase Avaki EII Overture*.

Viewing ACLs

Follow these steps to display the ACL for a file, a grid directory, or an Avaki share.









Step 1 Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>	 /	Attributes	Security	
<input type="checkbox"/>	 GeneratedViews	Attributes	Security	Categories
<input type="checkbox"/>	 Interconnects	Attributes	Security	Categories
<input type="checkbox"/>	 Metadata	Attributes	Security	Categories
<input type="checkbox"/>	 Shares	Attributes	Security	Categories
<input type="checkbox"/>	 System	Attributes	Security	Categories
<input type="checkbox"/>	 WSDLs	Attributes	Security	Categories
<input type="checkbox"/>	 Categories	Attributes	Security	Categories

Step 2 On the Browse Directories screen, click on a directory name (Shares, for example) to navigate to the file or directory whose ACL you want to see.

- Step 3** Click the **Security** link for the object you're interested in. The system displays the View Security Information screen, which includes the object's access control list (ACL) and other information.

View Security Information

You are viewing security information for the following object:
/System/Domains/Bedrock/Services/ShareManagers/Water Buffalo Lodge_1130096305646

The current owner of the object is the User **Administrator** in domain **Bedrock** and the Grid authentication service **DefaultAuthService**.

The following users and groups have been added to the Access Control List (ACL) for this object. To modify a user or group's permissions, place a check mark next to the user or group and click **Edit All Checked**. To add a user who is in the current grid domain, click **Add User to ACL**. To add a group that is in the current domain, click **Add Group to ACL**. To add a user or group that is in a connected domain, click **Add Via Interconnect ID**.

Select	Name	Type	Domain	Auth Service	Auth Service Type	Read	Write	Execute	Delete
<input type="checkbox"/>	Administrator	User	Bedrock	DefaultAuthService	Grid	Allow	Allow	Allow	Allow
<input type="checkbox"/>	wilma	User	Bedrock	DefaultAuthService	Grid	Allow	Allow	Allow	Allow

Manage Access Control Lists

[Edit All Checked](#) Configure permissions for each user or group in the ACL for an object

[Add User to ACL](#) Add a user to the ACL for an object

[Add Group to ACL](#) Add a group to the ACL for an object

[Add Via Interconnect ID](#) Upload an interconnect ID to add a user or group to the ACL for an object

Modifying permissions in ACLs

Follow these steps to change the read, write, execute, and delete permissions for a file, an Avaki directory, a data service, a database operation, or any other object in the data catalog. To modify the permissions on an object, you must be the owner of the object or a member of the Administrators group.

Note Members of the Administrators group can perform any action (read, write, execute, delete) on any object, regardless of how permissions are set. To minimize confusion, we recommend that you do not modify the administrator's permissions.









Step 1 Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>	 /	Attributes	Security	
<input type="checkbox"/>	 GeneratedViews	Attributes	Security	Categories
<input type="checkbox"/>	 Interconnects	Attributes	Security	Categories
<input type="checkbox"/>	 Metadata	Attributes	Security	Categories
<input type="checkbox"/>	 Shares	Attributes	Security	Categories
<input type="checkbox"/>	 System	Attributes	Security	Categories
<input type="checkbox"/>	 WSDLs	Attributes	Security	Categories
<input type="checkbox"/>	 Categories	Attributes	Security	Categories

Step 2 On the Browse Directories screen, click a directory name (Shares, for example) to navigate to the file or directory whose ACL you want to set.

- Step 3** Click the **Security** link for the object you're interested in. The system displays the View Security Information screen, which includes the ACL and other information.

View Security Information

You are viewing security information for the following object:
/System/Domains/Bedrock/Services/ShareManagers/Water Buffalo Lodge_1130096305646

The current owner of the object is the User **Administrator** in domain **Bedrock** and the Grid authentication service **DefaultAuthService**.

The following users and groups have been added to the Access Control List (ACL) for this object. To modify a user or group's permissions, place a check mark next to the user or group and click **Edit All Checked**. To add a user who is in the current grid domain, click **Add User to ACL**. To add a group that is in the current domain, click **Add Group to ACL**. To add a user or group that is in a connected domain, click **Add Via Interconnect ID**.

Select	Name	Type	Domain	Auth Service	Auth Service Type	Read	Write	Execute	Delete
<input type="checkbox"/>	Administrator	User	Bedrock	DefaultAuthService	Grid	Allow	Allow	Allow	Allow
<input type="checkbox"/>	wilma	User	Bedrock	DefaultAuthService	Grid	Allow	Allow	Allow	Allow

Manage Access Control Lists

[Edit All Checked](#) Configure permissions for each user or group in the ACL for an object
[Add User to ACL](#) Add a user to the ACL for an object
[Add Group to ACL](#) Add a group to the ACL for an object
[Add Via Interconnect ID](#) Upload an interconnect ID to add a user or group to the ACL for an object

- Step 4** In the Select column, click a check box to select the object whose permissions you want to change. (You can click more than one box.)

Step 5 Click **Edit All Checked**. The system displays the Modify Permissions screen.

Modify Permissions

You may specify read, write, execute, or delete permission for each user or group in the ACL for the following object:

/System/Domains/Bedrock/Services/ShareManagers/Water Buffalo Lodge_1130096305646

For each permission, you may specify the allow, deny, or unset option, or you may leave the current value as is. (If you choose the unset option for all permissions and the user is not the object's owner, the user will be removed from the ACL.)

Select the new permissions for the user or group:

Name	Type	Domain	Auth Service	Auth Service Type	Set as Owner	Delete	Permissions															
wilma	User	Bedrock	DefaultAuthService	Grid	<input type="checkbox"/>	<input type="checkbox"/>	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;"></th> <th style="width: 10%;">Current</th> <th style="width: 10%;">New</th> </tr> </thead> <tbody> <tr> <td>Read</td> <td>Allow</td> <td>As is ▾</td> </tr> <tr> <td>Write</td> <td>Allow</td> <td>As is ▾</td> </tr> <tr> <td>Execute</td> <td>Allow</td> <td>As is ▾</td> </tr> <tr> <td>Delete</td> <td>Allow</td> <td>As is ▾</td> </tr> </tbody> </table>		Current	New	Read	Allow	As is ▾	Write	Allow	As is ▾	Execute	Allow	As is ▾	Delete	Allow	As is ▾
	Current	New																				
Read	Allow	As is ▾																				
Write	Allow	As is ▾																				
Execute	Allow	As is ▾																				
Delete	Allow	As is ▾																				

Step 6 The table in the Modify Permissions screen lists users and groups on the left. For each user or group, you can:

- Click the box in the Set as Owner column to make this user or group the owner of the object.
- Click the box in the Delete column to remove this user or group from the ACL.
- Change a value under New in the Permissions column to reset the read, write, execute, or delete permissions for this user or group. Use the pull-down menus to select one of these values:
 - As is
Leaves the permission unchanged.
 - Unset
Indicates that this value has not been set. When a permission value is Unset, the user's permission for this action (read, write, execute, or delete) may depend on other permissions (group or individual) for this object. For example, if the user's own account has a permission of Unset for reading a file but the user belongs to a group that is allowed to read the file, the user is allowed to read the file. If the

user's account has a permission of Unset for reading a file and no group permissions apply, the user is not allowed to read the file.

- Deny
This user may not perform this action (read, write, execute, or delete) on the object. In cases where two or more permissions disagree (for example, a user belongs to a group that is allowed to read a file but the user himself is denied permission to read that file), Deny always wins. There is one exception: the everyone group. If “everyone” is in the allow list, denying an individual has no effect. Note that it is not possible to deny permission to a group, except in the case of the everyone group.
- Allow
This user or group may perform this action (read, write, execute, or delete) on the object, unless permission is denied elsewhere. (For example, if a user belongs to a group that is allowed to read a file, but the user himself is denied permission to read that file, the user is not able to read the file.)

- Step 7** If you are changing permissions for a directory, specify whether the changes apply only to the directory itself, or recursively to all its contents. Click either **Apply changes to selected directory only** or **Apply changes to selected directory and all contents**.
- Step 8** Click **Submit** to save your changes. The system displays the changed permissions on the View Security Information screen.

Adding users and groups to ACLs

Follow these steps to add a user or a group to the ACL for a file, a grid directory, or any other object in the data grid. Once you have added a user or group to an object's ACL, you can grant or deny read, write, execute, and delete permissions on the object.

Note When an Avaki user is added to a group, the user must log out and log back in to enjoy any additional access enabled by the new group membership.

Note See [“Enabling user or group access” on page 305](#) for a discussion of how to use interconnect IDs to add users and groups to ACLs in interconnected Avaki domains.

- Step 1** Navigate to the Browse Directories screen:
Home > Data catalog management

- Step 2** On the Browse Directories screen, click a directory name (Shares, for example) to navigate to the file or directory whose ACL you want to set.
- Step 3** Click the **Security** link for the object you're interested in. The system displays the View Security Information screen, which includes the ACL and other information.

View Security Information

You are viewing security information for the following object:
/System/Domains/Bedrock/Services/ShareManagers/Water Buffalo Lodge_1130096305646

The current owner of the object is the User **Administrator** in domain **Bedrock** and the Grid authentication service **DefaultAuthService**.

The following users and groups have been added to the Access Control List (ACL) for this object. To modify a user or group's permissions, place a check mark next to the user or group and click **Edit All Checked**. To add a user who is in the current grid domain, click **Add User to ACL**. To add a group that is in the current domain, click **Add Group to ACL**. To add a user or group that is in a connected domain, click **Add Via Interconnect ID**.

Select	Name	Type	Domain	Auth Service	Auth Service Type	Read	Write	Execute	Delete
<input type="checkbox"/>	Administrator	User	Bedrock	DefaultAuthService	Grid	Allow	Allow	Allow	Allow
<input type="checkbox"/>	wilma	User	Bedrock	DefaultAuthService	Grid	Allow	Allow	Allow	Allow

Manage Access Control Lists

[Edit All Checked](#) Configure permissions for each user or group in the ACL for an object

[Add User to ACL](#) Add a user to the ACL for an object

[Add Group to ACL](#) Add a group to the ACL for an object

[Add Via Interconnect ID](#) Upload an interconnect ID to add a user or group to the ACL for an object

- Step 4** Click **Add User to ACL** or **Add Group to ACL**. The system displays the Add Grid User to ACL screen, below, or the Add Grid Group to ACL screen, which is very similar.

Current Domain: Bedrock Current Server: ANTIMONYXP.sybase.com

Grid domain to add users from:

Add Grid User to ACL

All users

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

In the Add column, place a check mark next to the names of the user(s) you want to add to the access control list (ACL) for the following object:

/System/Domains/Bedrock/Services/ShareManagers/Water Buffalo Lodge_1130096305646

Add	Username	Authentication service	Authentication service type
<input type="checkbox"/>	Administrator	DefaultAuthService	Grid
<input type="checkbox"/>	MessagingUser	DefaultAuthService	Grid
<input type="checkbox"/>	barney	DefaultAuthService	Grid
<input type="checkbox"/>	betty	DefaultAuthService	Grid
<input type="checkbox"/>	fred	DefaultAuthService	Grid
<input type="checkbox"/>	wilma	DefaultAuthService	Grid

- Step 5** If you don't see the user or group you want to add, click on a letter to display users or groups beginning with that letter.
- Step 6** Check boxes in the Add column to select the users or groups you want to add to the ACL.
- Step 7** Click **Submit**. The system displays the object's ACL for the new user or group. For instructions on editing the ACL, see [“Modifying permissions in ACLs” on page 240](#).

Using attributes

Each object in the data catalog—each file, directory, shared directory, data service, SQL view, database operation, server, and service—has *attributes* that store information such as the time at which the object was created and the name of the user who owns the object. By default, each file, directory, and Avaki share has these system attributes:

- Creation time
- Grid server the file is shared on
- Last access time
- Last change time
- Link count (the number of contexts to which a link is bound)
- Owner
- Size

You can create new attributes for any Avaki object for which you have write permission. (We recommend against adding attributes in the /System directory.) Because you can search on attributes, you might find it useful to create attributes that contain search strings for files and directories that you'll want to find in a search. Or you might want to attach a project name or a date to certain files.

This section contains the following procedures:

- [“Viewing attributes” on page 247](#)
- [“Creating new attributes” on page 249](#)
- [“Setting attribute values” on page 253](#)
- [“Deleting attributes” on page 255](#)

Note For information on searching on attributes, see [“Searching the data catalog” on page 231](#).

Viewing attributes

Follow these steps to display the attributes for a particular grid file, directory, Avaki share, Avaki server, or service.









Step 1 Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>		/	Attributes	Security	
<input type="checkbox"/>		GeneratedViews	Attributes	Security	Categories
<input type="checkbox"/>		Interconnects	Attributes	Security	Categories
<input type="checkbox"/>		Metadata	Attributes	Security	Categories
<input type="checkbox"/>		Shares	Attributes	Security	Categories
<input type="checkbox"/>		System	Attributes	Security	Categories
<input type="checkbox"/>		WSDLs	Attributes	Security	Categories
<input type="checkbox"/>		Categories	Attributes	Security	Categories

Step 2 On the Browse Directories screen, click a directory name (Shares, for example) to navigate to the file, directory, or other object whose attributes you want to display.

- Step 3** Click the **Attributes** link to the right of the file or directory name. The system displays the attributes for the selected object.

View Attributes

/System/Domains/CherylDomain/MyProjects has the following attributes:

System Attributes		
Name	Type	Value
system/NodeIndShareServicePath	string	/System/LocalDomain/Services/ShareManagers/MyProjects_1059956512611
system/ObjectHostName	string	192.168.0.2:1099
system/ObjectHostJndiPort	string	1099
system/References	integer	1
system/ObjectLoid	string	loid://1/f6c8add434/f6c8add71c/f6ca54dbce
system/ObjectType	string	mapped folder
system/Owner	string	/System/Domains/CherylDomain/Services/AuthServices/Grid/DefaultAuthService/Users/Administrator
system/FileSize	integer	512
system/ChangeTime	timestamp	2003-08-03 19:21:54.253
system/ModificationTime	timestamp	2003-08-03 19:21:55.064
system/LastAccessTime	timestamp	2003-06-28 08:43:31.212
system/ShareServerName	string	local

User-defined Attributes

A user-defined attribute may be one of the following types:

Type	Description	Format
String	Text	Any characters
Integer	Any whole quantity	Any integer
Float	A numeric value that can be fractional or very large	Any numeric value
Date	Year, month, and date	yyyy-mm-dd
Time	Hour, minute and second that an event occurs	hh:mm:ss
Timestamp	A precise time and date	yyyy-mm-dd hh:mm:ss.ffffff

The screen shows two sets of attributes:

- **System attributes**
The attributes provided by default (see [page 246](#)). System attributes are read-only. Every object in the grid has system attributes, though the system attributes are not the same for every object.
- **User-defined attributes**
Read/write attributes that can be set by the owner of the files.

Creating new attributes

Follow these steps to create new attributes for a particular grid file, directory, Avaki share, server, or service.









Step 1 Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>		/	Attributes	Security
<input type="checkbox"/>		GeneratedViews	Attributes	Security Categories
<input type="checkbox"/>		Interconnects	Attributes	Security Categories
<input type="checkbox"/>		Metadata	Attributes	Security Categories
<input type="checkbox"/>		Shares	Attributes	Security Categories
<input type="checkbox"/>		System	Attributes	Security Categories
<input type="checkbox"/>		WSDLs	Attributes	Security Categories
<input type="checkbox"/>		Categories	Attributes	Security Categories

Step 2 On the Browse Directories screen, click a directory name (Shares, for example) to navigate to the file, directory, or other object for which you want to create a new attribute.

Step 3 Click the **Attributes** link to the right of the file or directory name. The system displays the attributes for the selected object.

View Attributes

/System/Domains/CherylDomain/MyProjects has the following attributes:

System Attributes		
Name	Type	Value
system/NodeIndShareServicePath	string	/System/LocalDomain/Services/ShareManagers/MyProjects_1059956512611
system/ObjectHostName	string	192.168.0.2:1099
system/ObjectHostJndiPort	string	1099
system/References	integer	1
system/ObjectLoid	string	loid://1/f6c8add434/f6c8add71c/f6ca54dbce
system/ObjectType	string	mapped folder
system/Owner	string	/System/Domains/CherylDomain/Services/AuthServices/Grid/DefaultAuthService/Users/Administrator
system/FileSize	integer	512
system/ChangeTime	timestamp	2003-08-03 19:21:54.253
system/ModificationTime	timestamp	2003-08-03 19:21:55.064
system/LastAccessTime	timestamp	2003-06-28 08:43:31.212
system/ShareServerName	string	local

User-defined Attributes		
A user-defined attribute may be one of the following types:		
Type	Description	Format
String	Text	Any characters
Integer	Any whole quantity	Any integer
Float	A numeric value that can be fractional or very large	Any numeric value
Date	Year, month, and date	yyyy-mm-dd
Time	Hour, minute and second that an event occurs	hh:mm:ss
Timestamp	A precise time and date	yyyy-mm-dd hh:mm:ss.ffffff

Step 4 To add a custom attribute at the bottom of the User-defined Attributes area, enter a name in the “Create new attribute” field. The name may be up to 255 characters long.

Step 5 Use the pull-down menu to select a type for the new attribute:

- **String:** Text, such as search strings or a project name. The string may be up to 255 characters long.
- **Integer:** Any whole quantity, such as the number of a particular component in a DNA strand. The valid range for integers is -9,223,372,036,854,775,808 to +9,223,372,036,854,775,807.

- **Float:** A numeric value that can be fractional or very large. This might be the load on a computer or a percentage, for example. The valid range for floats is $\pm 4.94065645841246544e-324$ to $\pm 1.79769313486231570e+308$.
- **Date:** Year, month, and date, such as a project date, file creation date, or a subject's birth date.
- **Time:** Hour, minute, and second that an event occurs. This can be a celestial event or flight schedule information, for example. The supported range is 00:00:00 to 23:59:59.
- **Timestamp:** A precise time (to fractions of a second) and date. This attribute type can be used to mark the start and end of an experiment or a chemical reaction, for example.

Step 6 Enter a value in the Value field. The formats for the date, time, and timestamp attribute types are as follows:

Type	Description	Format	Example
Date	Year, month, and date	yyyy-mm-dd	2002-09-21
Time	Hour, minute, and second	hh:mm:ss	04:05:06
Timestamp	Year, month, date, hour, minute, second, fraction of a second	yyyy-mm-dd hh:mm:ss.fffffff	2002-09-21 04:05:06.57567

Step 7 Click **Submit**. The new attribute is added to the User-defined Attributes table.

User-defined Attributes			
A user-defined attribute may be one of the following types:			
Type	Description	Format	
String	Text	Any characters	
Integer	Any whole quantity	Any integer	
Float	A numeric value that can be fractional or very large	Any numeric value	
Date	Year, month, and date	yyyy-mm-dd	
Time	Hour, minute and second that an event occurs	hh:mm:ss	
Timestamp	A precise time and date	yyyy-mm-dd hh:mm:ss.ffffff	
To create a new attribute, specify the attribute's name, type, and value. To delete an attribute, place a check mark next to it in the Delete column.			
Name	Type	Value	Delete
Project	string	<input type="text" value="Doc Project"/>	<input type="checkbox"/>
Create new attribute:	<input type="text"/> String	<input type="text"/>	
<input checked="" type="radio"/> Apply changes to selected directory only <input type="radio"/> Apply changes to selected directory and all contents			
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>			

Setting attribute values

Follow these steps to change the values of attributes for a particular grid file, directory, share, server, or service. You cannot modify system (default) attributes.









Step 1 Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>	 /	Attributes	Security
<input type="checkbox"/>	 GeneratedViews	Attributes	Security Categories
<input type="checkbox"/>	 Interconnects	Attributes	Security Categories
<input type="checkbox"/>	 Metadata	Attributes	Security Categories
<input type="checkbox"/>	 Shares	Attributes	Security Categories
<input type="checkbox"/>	 System	Attributes	Security Categories
<input type="checkbox"/>	 WSDLs	Attributes	Security Categories
<input type="checkbox"/>	 Categories	Attributes	Security Categories

Step 2 On the Browse Directories screen, click a directory name (Shares, for example) to navigate to the file, directory, or other object whose attributes you want to display.

Step 3 Click the **Attributes** link to the right of the file or directory name. The system displays the attributes for the selected object.

- Step 4** In the User-defined Attributes table, click in the field in the Name column and enter the name of the attribute to modify. Click in the field in the Value column and enter a new value for the attribute.

Name	Type	Value	Delete
Project	string	Doc Project	<input type="checkbox"/>
Create new attribute: Project	String	My Doc Project	

- Apply changes to selected directory only
 Apply changes to selected directory and all contents

- Step 5** Change the values of other attributes on this screen as needed.

- Step 6** Click **Submit**. The new attribute values are saved.

Deleting attributes

Follow these steps to delete user-defined attributes for a particular grid file, directory, Avaki share, server, or service. You cannot delete system (default) attributes.

Step 1 Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/> /	Attributes	Security
<input type="checkbox"/> GeneratedViews	Attributes	Security Categories
<input type="checkbox"/> Interconnects	Attributes	Security Categories
<input type="checkbox"/> Metadata	Attributes	Security Categories
<input type="checkbox"/> Shares	Attributes	Security Categories
<input type="checkbox"/> System	Attributes	Security Categories
<input type="checkbox"/> WSDLs	Attributes	Security Categories
<input type="checkbox"/> Categories	Attributes	Security Categories

Step 2 On the Browse Directories screen, click a directory name (Shares, for example) to navigate to the file, directory, or other object whose attributes you want to delete.

Step 3 Click the **Attributes** link to the right of the file or directory name. The system displays the attributes for the selected object.

- Step 4** In the User-defined Attributes area, click the check box in the Delete column for any attributes you want to remove.

Name	Type	Value	Delete
Project	string	Doc Project	<input checked="" type="checkbox"/>
Create new attribute: <input type="text"/>	String <input type="button" value="v"/>	<input type="text"/>	

- Apply changes to selected directory only
 Apply changes to selected directory and all contents

- Step 5** Click **Submit** to remove the selected attribute(s).

Managing Avaki shares

This chapter covers the following topics:

- “Overview of Avaki shares,” below
- “Creating Avaki shares” on page 258
- “Refreshing Avaki shares” on page 262
- “Adding share servers to Avaki shares” on page 263
- “Removing share servers from Avaki shares” on page 265
- “Changing the configuration of Avaki shares” on page 266
- “Uploading files to Avaki shares” on page 282
- “Moving shared data” on page 283
- “Taking Avaki shares on- and off-line” on page 285
- “Disconnecting Avaki shares” on page 288

Note Basic file and directory tasks, such as creating directories, moving files and directories, and performing searches, are covered in [Chapter 6, “Basic tasks”](#). These tasks are the same in Avaki shares and other Avaki directories.

Overview of Avaki shares

Shared directories, or Avaki shares, are links that allow directories in the local file system to appear in the Avaki data catalog. A data owner with a set of files can publish files by sharing the directory that contains them into a data grid, naming the directory in the data catalog structure, and specifying access rights. Applications and users who need the files can then access them through the data catalog without having to know which computer or file system they are located on.

Files that can be shared include files in direct-attached storage, network-attached storage, or storage area network storage. Avaki can handle very large (64-bit) files.

Sybase recommends that you share files from the data's local file system. While it's possible to share files from a mounted NFS or CIFS volume, this approach is less efficient because it adds one or more network hops.

Creating Avaki shares

To create an Avaki shared directory, you must be in the DataProviders or Administrators group. Follow these steps to share a local directory into your Avaki domain.

Note Any files or subdirectories to which you do not have read permission will not be included in the shared directory you create.

- Step 1** Log in to the grid domain as a member of the DataProviders or Administrators group.
- Step 2** Navigate to the Browse Directories screen:

Home > Data catalog management

Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

<input type="checkbox"/>	/	Attributes	Security	
<input type="checkbox"/>	GeneratedViews	Attributes	Security	Categories
<input type="checkbox"/>	Interconnects	Attributes	Security	Categories
<input type="checkbox"/>	Metadata	Attributes	Security	Categories
<input type="checkbox"/>	Shares	Attributes	Security	Categories
<input type="checkbox"/>	System	Attributes	Security	Categories
<input type="checkbox"/>	WSDLs	Attributes	Security	Categories
<input type="checkbox"/>	Categories	Attributes	Security	Categories

- Step 3** On the Browse Directories screen, click on directory names (Shares, for example) to navigate to the Avaki directory in which you want the new Avaki share to appear.

Step 4 Click **Create Avaki Share**. The Create Avaki Share screen appears.

Create Avaki Share

An Avaki share exports a directory from a local file system into the grid. To make local files available in the grid, you must explicitly share the directory containing those files.

To create an Avaki share, you must be an administrator or a member of the DataProviders group. When you create a share, you must specify a share name. A grid directory with this name is placed inside the parent directory you've selected. This subdirectory contains the shared data. The share name also appears on the [View Avaki Shares](#) screen, which lists the Avaki shares that exist in your grid domain.

If you delete or add a file to the share's grid directory, the change is propagated into the local file system immediately.

To change a share's owner, click **Select User** or **Select Group** and select a new owner.

Grid parent directory: /Shares

Share name:

Grid server:

Share server:

Local path on share server:

Encryption level:

Current owner: Administrator

New owner: [Select User](#) or [Select Group](#)

Description (optional):

Step 5 Fill in the form:

- **Share name:** Enter a name for the Avaki share service. The name can be up to 32 characters long. An Avaki directory with this name is placed inside the parent directory you've selected. This subdirectory contains the shared data. The share name also appears on the View Avaki Shares screen, which lists the Avaki shares in your Avaki domain.
- **Grid server:** From the pull-down menu, select the Avaki grid server on which the share will reside. (The shared files need not be located on the same machine as the grid server, but in that case we recommend that you set up a share server local to the shared files.)

- **Share server:** If you want the new Avaki share to use a particular share server, select it from the pull-down menu. If you leave the pull-down menu set to “Local to grid server,” the new share will use the grid server you specified in the “Grid server” field as its share server. You can configure additional share servers for this share later—see [“Adding share servers to Avaki shares” on page 263](#).
- **Local path on share server:** The path to the local directory that you are sharing into the data grid. (This file system should be local to the share server, or to the grid server if you chose “Local to grid server” above.) For example:

```
C:\testdir
```

- **Encryption level:** From the pull-down menu, choose **Clear** (encryption is turned off for this Avaki share) or **Encrypted** (encryption is turned on).

All communication between an Avaki share and its grid server is encrypted. The encryption level property lets you, the data provider, decide whether your shared data also needs to be encrypted when it travels over the wire.

- **New owner (optional):** By default, new Avaki shares are owned by the Administrator user. To change an Avaki share’s owner, click **Select User** or **Select Group** and select a new owner.
- **Description (optional):** Enter some descriptive information about this Avaki share.

Step 6 Click **Submit** to create the shared directory. The share service processes files at a rate of about 200 files per second, so the creation of an Avaki share that contains many files can take several minutes. When the process is complete, the system redisplay the Browse Directories screen, showing the new Avaki share.

Step 7 Consider configuring the following aspects of your new Avaki share:

- **Refresh schedule**
The refresh schedule determines how often the Avaki share rereads its source directory. There is no default refresh schedule. See [“Changing the refresh schedule” on page 266](#).
- **Share servers**
To improve the availability of your shared data, you can add one or more share servers to an Avaki share. See [“Adding share servers to Avaki shares” on page 263](#).
- **Permissions**
By default, the permissions on a new shared directory are set so that the user who created the share (who is the owner) can read, write, execute, and delete it and its contents; members of the DomainUsers group in the current grid domain can read

the share and its contents, but they cannot write, execute, or delete. Users in other domains have no permissions. For instructions on configuring permissions (access control lists), see [“Using access control lists” on page 237](#).

Refreshing Avaki shares

When an Avaki share refreshes (rehashes), it rereads the contents of its shared directory from the local file system. Often, it makes sense to configure your Avaki shares to refresh automatically at scheduled intervals. (For instructions, see [“Changing the refresh schedule” on page 266](#).) In certain situations, however, you need to force a manual refresh—when you know the source directory has changed, for example. Follow these steps to force an immediate refresh of an Avaki share.

Note A large Avaki share takes some time to complete a refresh—the rate is about 200 files per second.

Step 1 Log in to the Avaki domain as a user who has write permission on the target share.

Step 2 Navigate to the View Avaki Shares screen:

Home > Service management > View Avaki shares

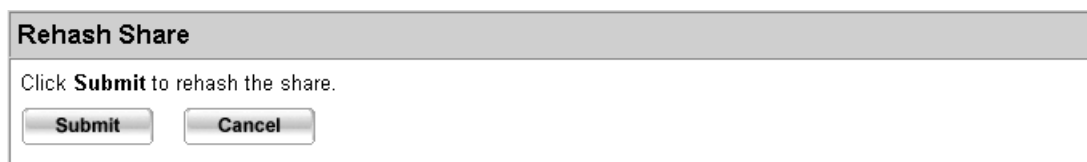
View Avaki Shares

The following Avaki shares have been integrated into the local grid domain. To shut down a share, click **Disconnect**. The share will be removed from the grid directory system, but the corresponding files in the underlying file system will not be deleted.

Avaki Share Name	Creation Date	View/Edit Share	Schedule	Shut Down	Attributes	Security
Bam-bamsfiles	Tue Sep 21 23:40:53 EDT 2004	View/Edit	Schedule	Disconnect	Attributes	Security
Barneysfiles	Tue Sep 21 23:37:33 EDT 2004	View/Edit	Schedule	Disconnect	Attributes	Security
Bettysfiles	Tue Sep 21 23:38:07 EDT 2004	View/Edit	Schedule	Disconnect	Attributes	Security

Step 3 Click the **View/Edit** link for the share you want to refresh.

Step 4 On the screen that appears, scroll down to the Rehash Share window.



Step 5 Click **Submit** to refresh (rehash) the Avaki share. The system reports “Share successfully rehashed” when the operation is finished.

Adding share servers to Avaki shares

In some cases, it makes sense to have multiple share servers serving a single Avaki share. When data is stored in network attached storage (NAS) or a storage area network (SAN), it is possible for multiple machines to have equivalent access to the same data. Multiple machines with equivalent access to the same data can provide equivalent access to Avaki shares by running share servers. If one share server fails or becomes unresponsive, another share server takes over transparently. An Avaki share with a single share server can be a performance bottleneck and a single point of failure; multiple share servers can reduce these problems significantly.

Before you can add a share server to an Avaki share, you must set up the share server and connect it to the same grid server on which your Avaki share resides. For instructions, see [“Setting up share servers” on page 54](#).

Prerequisites for configuring several share servers on a single Avaki share:

- Each share server machine must have an equivalent view of the data in the share.
- The local file system path to the root of the data being accessed must be the same across all share server machines.
- The Avaki share and all its share servers must be connected to the same grid server.
- You must have read and write permission on all share servers in the configuration.

Note To provide the best performance, the share server you add should be on a different machine from the other share server(s) serving this Avaki share.

Follow these steps to add an existing share server to an existing Avaki share.

Step 1 Log in to the grid domain as a user who has write permission on the Avaki share to which you want to add a share server.

Step 2 Navigate to the View Avaki Shares screen:

Home > Service management > View Avaki shares

View Avaki Shares						
The following Avaki shares have been integrated into the local grid domain. To shut down a share, click Disconnect . The share will be removed from the grid directory system, but the corresponding files in the underlying file system will not be deleted.						
Avaki Share Name	Creation Date	View/Edit Share	Schedule	Shut Down	Attributes	Security
Bam-bamsfiles	Tue Sep 21 23:40:53 EDT 2004	View/Edit	Schedule	Disconnect	Attributes	Security
Barneysfiles	Tue Sep 21 23:37:33 EDT 2004	View/Edit	Schedule	Disconnect	Attributes	Security
Bettysfiles	Tue Sep 21 23:38:07 EDT 2004	View/Edit	Schedule	Disconnect	Attributes	Security
<input type="button" value="Done"/>						

Step 3 Click View/Edit for the share to which you are adding a share server.

Step 4 On the screen that appears, scroll down to the Update Share Servers window and click **Share servers**. The View Servers for Share screen appears.

View Servers for Share			
You are viewing the share servers for Water Buffalo Lodge .			
Server Name	Current LBF	New LBF	Delete
local	1	<input type="text" value="1"/>	<input type="checkbox"/>
<input type="button" value="Submit"/> <input type="button" value="Cancel"/> <input type="button" value="Add Server"/>			

Step 5 Click the **Add Server** button. The Add Server to Share screen appears.

Add Server To Share	
You are viewing the share servers connected to the grid server on which Water Buffalo Lodge resides.	
Select	Share Server Name
<input type="checkbox"/>	local
<input type="checkbox"/>	standaloneshareserv
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>	

- Step 6** The Add Server to Share screen lists share servers connected to the grid server on which your Avaki share resides. Click a box in the Select column to choose a share server to associate with your share. Then click **Submit**.
- Step 7** The View Servers for Share screen reappears, now displaying both the old and the new share servers.

For instructions on setting the LBF, see [“Changing the load balancing factor” on page 280](#).

Removing share servers from Avaki shares

Follow these steps to remove a share server from an Avaki share.

Note You cannot use this procedure to remove the only share server associated with an Avaki share. To disconnect an Avaki share, see [“Taking Avaki shares on- and off-line” on page 285](#) or [“Disconnecting Avaki shares” on page 288](#).

- Step 1** Log in to the grid domain as a user who has write permission on the Avaki share from which you want to remove a share server.
- Step 2** Navigate to the View Avaki Shares screen:

Home > Service management > View Avaki shares

View Avaki Shares

The following Avaki shares have been integrated into the local grid domain. To shut down a share, click **Disconnect**. The share will be removed from the grid directory system, but the corresponding files in the underlying file system will not be deleted.

Avaki Share Name	Creation Date	View/Edit Share	Schedule	Shut Down	Attributes	Security
Bam-bamsfiles	Tue Sep 21 23:40:53 EDT 2004	View/Edit	Schedule	Disconnect	Attributes	Security
Barneysfiles	Tue Sep 21 23:37:33 EDT 2004	View/Edit	Schedule	Disconnect	Attributes	Security
Bettysfiles	Tue Sep 21 23:38:07 EDT 2004	View/Edit	Schedule	Disconnect	Attributes	Security

- Step 3** Click View/Edit for the share whose share server you want to remove.

- Step 4** On the screen that appears, scroll down to the Update Share Servers window and click **Share servers**. The View Servers for Share screen appears.

View Servers for Share

You are viewing the share servers for **Water Buffalo Lodge**.

Server Name	Current LBF	New LBF	Delete
local	1	<input style="width: 100px;" type="text" value="1"/>	<input type="checkbox"/>
standaloneshareserv	1	<input style="width: 100px;" type="text" value="1"/>	<input type="checkbox"/>

- Step 5** Click the box in the Delete column for the share server you want to remove from the share. Then click **Submit**. The system redisplay the View Servers for Share screen to show that the share server you selected has been removed.

Changing the configuration of Avaki shares

After you create an Avaki share, you can change these properties:

- The refresh schedule—see [“Changing the refresh schedule,”](#) below
- The encryption level—see [“Changing the encryption level”](#) on page 279
- The load balancing factor (LBF)—see [“Changing the load balancing factor”](#) on page 280

You can also change the names of files and subdirectories within a share; see [“Managing files and directories”](#) on page 208. Note that you cannot change the name of the Avaki share itself.

Changing the refresh schedule

The refresh (sometimes called rehash) schedule determines when an Avaki share rereads the contents of its source directory from the local file system. Refreshing picks up directory-level changes: files and subdirectories that have been added, deleted, moved or renamed. By default, no refreshes are scheduled—an Avaki share will never be updated unless you create a refresh schedule or force a refresh.

In addition to scheduling refreshes, you can configure *schedule exclusions*—times when refreshes or other scheduled events are not allowed to occur.

This section includes three procedures:

- [“Adding entries to the refresh schedule,”](#) below
- [“Configuring schedule exclusions”](#) on page 274
- [“Removing entries from the refresh schedule”](#) on page 278

Adding entries to the refresh schedule

To configure the refresh schedule for an Avaki share, you must have write permission on the share.

There are four types of schedules: one-time, periodic, calendared, and advanced. You configure one-time, periodic, and calendared schedules using the UI’s graphical tools. An advanced schedule is similar to a calendared schedule, except that in an advanced schedule you must supply a cron expression to specify the recurrence interval. Cron expressions provide more flexibility in specifying times. For example, they allow you to schedule to the precise minute and second (the UI offers 5-minute granularity), and they allow you to schedule two or more times in one schedule entry (such as 5 am and 6 pm every Tuesday) where the UI allows only one. For information on the syntax of cron expressions, see the *Sybase Avaki EII Command Reference*.

Follow these steps to add entries to the refresh schedule for an Avaki share. You can create multiple entries for one share if you wish.

- Step 1** To display a list of Avaki shares in the data catalog, navigate to the View Avaki Shares screen:

Home > Service management > View Avaki shares.

View Avaki Shares						
The following Avaki shares have been integrated into the local grid domain. To shut down a share, click Disconnect . The share will be removed from the grid directory system, but the corresponding files in the underlying file system will not be deleted.						
Avaki Share Name	Creation Date	View/Edit Share	Schedule	Shut Down	Attributes	Security
Bam-bamsfiles	Tue Sep 21 23:40:53 EDT 2004	View/Edit	Schedule	Disconnect	Attributes	Security
Barneysfiles	Tue Sep 21 23:37:33 EDT 2004	View/Edit	Schedule	Disconnect	Attributes	Security
Bettysfiles	Tue Sep 21 23:38:07 EDT 2004	View/Edit	Schedule	Disconnect	Attributes	Security

- Step 2** Click the **Schedule** link for the Avaki share you want to modify.
- Step 3** On the screen that appears, click **Add schedule**. The system displays the Add New Schedule screen.

Add New Schedule
Select a time zone for this schedule: (server's local time zone) ▼
Type of schedule:
<input type="button" value="One time"/> <input type="button" value="Periodic"/> <input type="button" value="Calendared"/> <input type="button" value="Advanced"/>
Starting <input checked="" type="radio"/> Now <input type="radio"/> 12 AM ▼ :00 ▼ Jan ▼ 1 ▼ 2005 ▼
Recur every: <input type="text"/> minutes ▼
Continue recurring
<input checked="" type="radio"/> forever <input type="radio"/> at most <input type="text"/> times <input type="radio"/> until 12 AM ▼ :00 ▼ Jan ▼ 1 ▼ 2005 ▼
Schedule exclusions:
No schedule exclusions are currently defined in this domain.
<input type="button" value="Submit"/> <input type="button" value="Cancel"/>

- Step 4** Click a tab to choose the type of schedule you want: One time, Periodic, Calendared, or Advanced.
- Step 5** Go to the appropriate procedure to complete your schedule entry:
- “Configuring one-time refresh schedules,” below
 - “Configuring periodic refresh schedules” on page 270
 - “Configuring calendared refresh schedules” on page 271
 - “Configuring advanced refresh schedules” on page 273

Configuring one-time refresh schedules.

Add New Schedule

Select a time zone for this schedule: (server's local time zone) ▼

Type of schedule:

One time
 Periodic
 Calendared
 Advanced

Do once: Now

12 AM ▼ :00 ▼ Jan ▼ 1 ▼ 2005 ▼

Follow these steps:

- Step 1** Choose a time zone for this schedule. If the people using this Avaki share are not in the same time zone as the share server or grid server that hosts the share, this option lets you set the schedule according to the users’ local time.
- Step 2** In the Do once field:
- If you want the one-time refresh to occur immediately, click the **Now** button.
 - If you want the refresh to occur later, click the lower button. Then use the pull-downs to select the time, month, day and year.
- Step 3** Click **Submit** to save your schedule entry. The system displays the new entry on the Show Share Rehash Schedules screen.

Note For instructions on setting up schedule exclusions—specific times when a scheduled activity such as refreshing an authentication service or an Avaki share

does not occur according to the schedule—see “Configuring schedule exclusions” on page 274.

Configuring periodic refresh schedules.

Add New Schedule

Select a time zone for this schedule: (server's local timezone) ▾

Type of schedule:

One time
 Periodic
 Calendared
 Advanced

Starting Now
 12 AM ▾ :00 ▾ Jan ▾ 1 ▾ 2005 ▾

Recur every: minutes ▾

Continue recurring

forever

at most times

until 12 AM ▾ :00 ▾ Jan ▾ 1 ▾ 2005 ▾

Follow these steps:

- Step 1** Choose a time zone for this schedule. If the people using this Avaki share are not in the same time zone as the share server or grid server that hosts the share, this option lets you set the schedule according to the users' local time.
- Step 2** In the Starting field, specify when this schedule should start:
 - If you want the first refresh to start immediately, click the **Now** button.
 - If you want the schedule to start later, click the lower button. Then use the pull-downs to select the time, month, day and year.
- Step 3** Use the Recur every field to specify the interval at which this schedule is executed. Enter an integer and select from the pull-down to specify an interval—for example, every 40 minutes, every 5 days, or every 2 months.
- Step 4** In the Continue recurring field, specify how long you want this schedule entry to remain in effect: forever, for a specified number of refreshes, or until a specified date and time.
- Step 5** Click **Submit** to save your schedule entry. The system displays a summary of the new entry, including the time of next execution, on the Show Share Refresh Schedules screen.

Note For instructions on setting up schedule exclusions—specific times when a scheduled activity such as refreshing an authentication service or an Avaki share does not occur according to the schedule—see [“Configuring schedule exclusions” on page 274](#).

Configuring calendared refresh schedules.

Add New Schedule

Select a time zone for this schedule: (server's local time zone) ▼

Type of schedule:

Now
 After 12 AM :00 Jan 1 2005

Recur at: 12 AM :00 on:

Days	Months	Years
<input checked="" type="radio"/> all <input type="radio"/> of week <input type="radio"/> of month <input type="radio"/> of week in month	Jan ▲ Feb ▲ Mar ▲ Apr ▲ May ▲ Jun ▲ Jul ▲ Aug ▲ Sep ▲ Oct ▲ Nov ▲ Dec ▲ all ▼	2004 ▲ 2005 ▲ 2006 ▲ 2007 ▲ 2008 ▲ 2009 ▲ 2010 ▲ 2011 ▲ 2012 ▲ 2013 ▲ 2014 ▲ 2015 ▲ all ▼

Continue recurring

forever
 at most times
 until 12 AM :00 Jan 1 2005

Follow these steps:

- Step 1** Choose a time zone for this schedule. If the people using this Avaki share are not in the same time zone as the share server or grid server that hosts the share, this option lets you set the schedule according to the users' local time.

- Step 2** In the After field, specify when this schedule entry takes effect:
- If you want the schedule to start immediately, click the **Now** button.
 - If you want the schedule to start later, click the lower button. Then use the pull-downs to select the time, month, day and year.
- Step 3** In the Recur at field, use the pull-downs to specify the time of day at which you want the refresh to take place. (If you want the share to refresh more than once a day, you can use a periodic or advanced schedule, or you can create separate schedule entries for the other refreshes.)
- Step 4** In the Days column, choose how you want to specify days in this schedule entry:
- All: every day.
 - Of week: Sunday through Saturday—click one or more days.
 - Of month: 1, 2, 3...—click one or more days.
 - Of week in month: use the pull-downs to choose the first, second, third, fourth, fifth, or last occurrence of any day of the week (the first Monday, for example).
- Step 5** In the Months column, select one or more months during which this schedule entry will be in effect, or select **all** for all months. Use Shift-click or Control-click to select multiple months.
- Step 6** In the Years column, select one or more years during which this schedule entry will be in effect, or select **all** for all years. Use Shift-click or Control-click to select multiple years.
- Step 7** In the Continue recurring field, specify how long you want this schedule entry to remain in effect: forever, for a specified number of refreshes, or until a specified date and time.
- Step 8** Click **Submit** to save your schedule entry. The system displays a summary of the new entry, including the time of next execution, on the Show Share Rehash Schedules screen.

Note For instructions on setting up schedule exclusions—specific times when a scheduled activity such as refreshing an authentication service or an Avaki share does not occur according to the schedule—see [“Configuring schedule exclusions” on page 274](#).

Configuring advanced refresh schedules. See [page 267](#) for information on choosing the advanced schedule option.

Add New Schedule

Select a time zone for this schedule: (server's local time zone) ▼

Type of schedule:

Now
 After 12 AM ▼ :00 ▼ Jan ▼ 1 ▼ 2005 ▼

Recur according to this cron expression:

Continue recurring

forever
 at most times
 until 12 AM ▼ :00 ▼ Jan ▼ 1 ▼ 2005 ▼

Follow these steps:

- Step 1** Choose a time zone for this schedule. If the people using this Avaki share are not in the same time zone as the share server or grid server that hosts the share, this option lets you set the schedule according to the users' local time.
- Step 2** In the After field, specify when this schedule entry takes effect:
- If you want the schedule to start immediately, click the **Now** button.
 - If you want the schedule to start later, click the lower button. Then use the pull-downs to select the time, month, day and year.
- Step 3** Use the Recur... field to specify the interval at which this schedule is executed. Enter a cron expression of this form:
- ```

<seconds> <minutes> <hours> <days-of-month> <months>
<days-of-week> [<years>]
```
- See the *Sybase Avaki EII Command Reference* for details of the cron syntax.
- Step 4** In the Continue recurring field, specify how long you want this schedule entry to remain in effect: forever, for a specified number of refreshes, or until a specified date and time.

- Step 5** Click **Submit** to save your schedule entry. The system displays a summary of the new entry, including the time of next execution, on the Show Share Rehash Schedules screen.

## Configuring schedule exclusions

A schedule exclusion is a named time period that you define. When you apply a schedule exclusion to an entry in a schedule, the exclusion prevents the scheduled activity—rereading an Avaki share from its source file system or running a data service to refresh the cached copy of its results, for example—from occurring during the time specified by the exclusion. A schedule exclusion can be applied to as many schedules as you like, and it can be applied to schedules for any scheduled activity, including refreshing Avaki shares, refreshing imported user accounts, and caching files, directories, and the results of database operations, data services, and generated views.

For example, suppose you have a large Avaki share that's scheduled to refresh once a day. This works well most of the time, but on the last day of the month, demand on the network (or the host computer) is very high and you want to reduce traffic. You can set up a schedule exclusion for the last day of every month and apply it to the share's refresh schedule. If necessary, you can apply the same exclusion to other schedules to further reduce traffic. You can also configure each schedule so that the scheduled activity occurs before or after the schedule exclusion period, or not all.

This section includes these procedures:

- [“Setting up schedule exclusions,”](#) below
- [“Applying schedule exclusions to schedule entries”](#) on page 276

**Setting up schedule exclusions.** To define a schedule exclusion, you must be a member of the DataProviders group. Only the owner of an exclusion can edit or delete it (unless the owner or an administrator has granted write permissions to other users).

Follow these steps to define a schedule exclusion.

- Step 1** Navigate to the Add Schedule Exclusion screen:

Home > Service management > Create schedule exclusion.

**Add Schedule Exclusion**

Name:

Description (optional):

Time Zone:

Type of schedule exclusion:

Entire day  
 From  : to  :

|    |    |    |    |    |    |    |    |    |    |    |      |
|----|----|----|----|----|----|----|----|----|----|----|------|
| 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10 |    |      |
| 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |    |      |
| 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | last |

**Step 2** In the Name field, enter a name to identify this schedule exclusion. (Later, when you apply this exclusion to a schedule entry, you'll select this name from a list of exclusions.)

**Step 3** (Optional) In the Description field, enter a description of this schedule exclusion.

**Step 4** From the Time Zone pull-down, select the time zone for this exclusion period. (This is the time zone that determines when the exclusion occurs; the exclusion will be in effect in all time zones.)

**Step 5** Under Type of schedule exclusion, click a tab to specify how often you want the exclusion to recur:

- **Daily:** the exclusion blocks scheduled activities during the specified period every day. Use the pull-downs in the From and to fields to set the hour and minute at which the exclusion period begins and ends.
- **Weekly:** the exclusion blocks scheduled activities during the specified period on the specified days of every week. To set the duration of the exclusion, click the Entire day button or click the lower button, then use the pull-downs in the From and to fields to set the hour and minute at which the exclusion period begins and ends.

To set the days of the week, click the boxes for one or more days.

Sybase Avaki EII Administration Guide

275

- **Monthly:** the exclusion blocks scheduled activities during the specified period on the specified days of every month. To set the duration of the exclusion, click the Entire day button or click the lower button, then use the pull-downs in the From and to fields to set the hour and minute at which the exclusion period begins and ends.

To set the days of the month, click the boxes for one or more days.

- **Yearly:** the exclusion blocks scheduled activities during the specified period on the specified days of every year. To set the duration of the exclusion, click the Entire day button or click the lower button, then use the pull-downs in the From and to fields to set the hour and minute at which the exclusion period begins and ends.

To set the days of the year, use the month and date pull-downs. Click **Add day** to add as many days as you need.

- **Custom:** the exclusion blocks scheduled activities during one or more periods that you define using the time, month, day, and year pull-downs. Click **Add new range** if you want to define additional time periods. Use the Custom tab to configure one-time exclusions.

**Step 6** Click **Submit** to save your schedule exclusion.

The procedure that follows explains how to incorporate schedule exclusions into schedules.

**Applying schedule exclusions to schedule entries.** Before you can follow this procedure, at least one schedule exclusion must already be configured. See “[Setting up schedule exclusions](#),” above, for instructions.

To apply a schedule exclusion, you must be a member of the DomainUsers group in the domain where the exclusion was created, and you must have write permission on the object (the Avaki share, for example) to which you are applying the schedule.

Follow these steps to apply a schedule exclusion.

**Step 1** Navigate to the View Avaki Shares screen:



Home > Service management > View Avaki Shares.

**View Avaki Shares**

The following Avaki shares have been integrated into the local grid domain. To shut down a share, click **Disconnect**. The share will be removed from the grid directory system, but the corresponding files in the underlying file system will not be deleted.

| Avaki Share Name | Creation Date                | View/Edit Share           | Schedule                 | Shut Down                  | Attributes                 | Security                 |
|------------------|------------------------------|---------------------------|--------------------------|----------------------------|----------------------------|--------------------------|
| Barn-barnsfiles  | Tue Sep 21 23:40:53 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Barneysfiles     | Tue Sep 21 23:37:33 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Bettysfiles      | Tue Sep 21 23:38:07 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

- Step 2** Click the **Schedule** link for the Avaki share whose refresh schedule you want to modify. The system displays the Show Share Refreshes screen.
- Step 3** Click **View/Edit** for the schedule you want to modify. (If there are no schedule entries yet, or if you want to add another one, click **Add Schedule**. See [“Adding entries to the refresh schedule” on page 267](#) for instructions on filling out the top part of the Add New Schedule screen, then return to this procedure to apply exclusions.)
- Step 4** Scroll down, if necessary, to expose the Schedule Exclusions portion of the View/Edit Schedule screen.

Exclusions:

Allow these exclusions to block this schedule: 3\_times\_per\_month End\_of\_month

When an exclusion conflicts with a scheduled occurrence, the system should:

cancel the occurrence

reschedule

When rescheduling, examine times before the conflicting exclusion, at increments of  

minutes .

- Step 5** Select a schedule exclusion from the Allow these exclusions... list. Use the up and down arrows to scroll if you don't see the exclusion you want.
- Step 6** Click to specify what should happen when an exclusion prevents this Avaki share from refreshing on schedule:
- **Cancel the occurrence** causes the share to skip any scheduled refreshes that fall within the exclusion period.

- **Reschedule** causes the system to reschedule any scheduled refreshes that fall within the exclusion period.

**Step 7** If you chose **Reschedule**, use the pull-downs in the bottom line to specify:

- Whether the system should reschedule the refresh for **before** or **after** the exclusion period
- How long before or after the exclusion period the system should try to reschedule the refresh

For example, if you specify that the share should refresh 2 hours before the exclusion period begins, the system first tries to reschedule the refresh at the 2 hour point. If that time slot isn't available because of another exclusion, the system tries to reschedule the refresh for 2 hours before the beginning of the other exclusion period, and so on.

**Step 8** Click **Submit** to apply the specified exclusion and rescheduling policy to the refresh schedule. The system displays the Show Share Rehash Schedule screen.

## Removing entries from the refresh schedule

To remove entries from the refresh schedule for an Avaki share, you must have write permission on the share. Follow these steps to remove entries from the refresh schedule.

**Step 1** To display a list of the local directories that have been shared into the data catalog, navigate to the View Avaki Shares screen:

Home > Service management > View Avaki shares.

**View Avaki Shares**

The following Avaki shares have been integrated into the local grid domain. To shut down a share, click **Disconnect**. The share will be removed from the grid directory system, but the corresponding files in the underlying file system will not be deleted.

| Avaki Share Name | Creation Date                | View/Edit Share           | Schedule                 | Shut Down                  | Attributes                 | Security                 |
|------------------|------------------------------|---------------------------|--------------------------|----------------------------|----------------------------|--------------------------|
| Barn-bamsfiles   | Tue Sep 21 23:40:53 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Barneysfiles     | Tue Sep 21 23:37:33 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Bettysfiles      | Tue Sep 21 23:38:07 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

- Step 2** Click **Schedule** beside the Avaki share you want to modify. The system displays the Show Share Rehash Schedules screen.

**Show Share Rehash Schedules**

This share is automatically rehashed according to these schedules:

| Select                   | Summary                                                        | View/Edit                 |
|--------------------------|----------------------------------------------------------------|---------------------------|
| <input type="checkbox"/> | Every 12 hours, next execution Wed Sep 22 12:21:26 AM EDT 2004 | <a href="#">View/Edit</a> |

- Step 3** Click the box in the Select column next to the schedule entry you want to remove.
- Step 4** Click **Delete All Checked** to save your changes. The system displays the how Share Rehash Schedules screen without the entry you deleted.

## Changing the encryption level

All communication between an Avaki share and its grid server is encrypted. The encryption level property lets you, the data provider, decide whether your shared data also needs to be encrypted when it travels over the wire.

Follow these steps to change the encryption level for a shared directory.

- Step 1** Navigate to the View Avaki Shares screen:

Home > Service management > View Avaki shares

**View Avaki Shares**

The following Avaki shares have been integrated into the local grid domain. To shut down a share, click **Disconnect**. The share will be removed from the grid directory system, but the corresponding files in the underlying file system will not be deleted.

| Avaki Share Name | Creation Date                | View/Edit Share           | Schedule                 | Shut Down                  | Attributes                 | Security                 |
|------------------|------------------------------|---------------------------|--------------------------|----------------------------|----------------------------|--------------------------|
| Bam-bamsfiles    | Tue Sep 21 23:40:53 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Barneysfiles     | Tue Sep 21 23:37:33 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Bettysfiles      | Tue Sep 21 23:38:07 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

- Step 2** Click **View/Edit** beside the Avaki share you want to modify. The Update Avaki Share Information window appears.

**Update Avaki Share Information**

You are viewing information about the selected share. To update the information, enter the new values and click **Submit**. Note that you may not update the share's local path unless the share is offline.

**Share name:** Barneysfiles  
**Creation date:** Tue Sep 21 23:37:33 EDT 2004  
**Grid server:** localhost:1099  
**Grid path:** /Shares/Barneysfiles  
**Local path:** C:\Documents and Settings\bthoene  
**Encryption level:**  Clear  Encrypted  
**Status:**  Online  Offline  
**Description:**

**Submit** **Cancel**

- Step 3** In the Encryption level field, choose **Clear** (encryption is turned off for this share) or **Encrypted** (encryption is turned on).
- Step 4** Click **Submit** to save your changes. The system redisplay the screen with the changed encryption level.

## Changing the load balancing factor

The load balancing factor (LBF) controls how requests for data from an Avaki share are distributed among two or more share servers associated with that share. The LBF has no effect on an Avaki share that has only one share server.

The approximate proportion of the load carried by a share server is a fraction defined as follows:

- The numerator is the LBF—an integer that specifies the portion of load that a share server owns for a given Avaki share.
- The denominator is the sum of the LBFs for all the share servers serving this Avaki share.

For example, suppose you have an Avaki share with three share servers, Larry, Curly, and Moe. You have configured LBFs for these share servers of 2, 1 and 1 respectively.

As a result, the share server Larry serves about 2/4, or 50%, of the requests for this Avaki share; Curly and Moe serve about 1/4 or 25% each.

The default LBF setting is 1. This means that if you don't alter the LBF values, each of an Avaki share's share servers will carry an equal portion of the load.

A share server that has been assigned to service a client's request to access a file will continue to serve any requests for that file. When the client requests another file, the LBF influences the selection of a share server to serve that new file.

Follow these steps to set the LBF for a share server.

**Step 1** Navigate to the View Avaki Shares screen:

Home > Service management > View Avaki shares

### View Avaki Shares

The following Avaki shares have been integrated into the local grid domain. To shut down a share, click **Disconnect**. The share will be removed from the grid directory system, but the corresponding files in the underlying file system will not be deleted.

| Avaki Share Name | Creation Date                | View/Edit Share           | Schedule                 | Shut Down                  | Attributes                 | Security                 |
|------------------|------------------------------|---------------------------|--------------------------|----------------------------|----------------------------|--------------------------|
| Barn-bamsfiles   | Tue Sep 21 23:40:53 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Barneysfiles     | Tue Sep 21 23:37:33 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Bettysfiles      | Tue Sep 21 23:38:07 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

**Step 2** Click **View/Edit** beside the Avaki share you want to modify.

**Step 3** On the screen that appears, scroll down to the Update Share Servers window and click **Share servers**. The system displays the View Servers for Share screen.

### View Servers for Share

You are viewing the share servers for **Water Buffalo Lodge**.

| Server Name         | Current LBF | New LBF                                              | Delete                   |
|---------------------|-------------|------------------------------------------------------|--------------------------|
| local               | 1           | <input style="width: 100px;" type="text" value="1"/> | <input type="checkbox"/> |
| standaloneshareserv | 1           | <input style="width: 100px;" type="text" value="1"/> | <input type="checkbox"/> |

**Step 4** In the New LBF column, enter new LBF values for one or more share servers.

- Step 5** Click **Submit** to save your changes. The system redisplay the View Servers for Share screen with your new value(s) in the Current LBF column.

## Uploading files to Avaki shares

Follow these steps to upload a file from the local file system to an Avaki share in the data catalog. Uploading a file to an Avaki share adds the file to the share's local source directory.

**Caution** When you upload a file to an Avaki share, the change is reflected in the source file system—not in the grid only.









- Step 1** Navigate to the Browse Directories screen:

Home > Data catalog management

**Browse Directories**

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

| <input type="checkbox"/> |  |                | <a href="#">Attributes</a> | <a href="#">Security</a> |                            |
|--------------------------|-------------------------------------------------------------------------------------|----------------|----------------------------|--------------------------|----------------------------|
| <input type="checkbox"/> | /                                                                                   |                | <a href="#">Attributes</a> | <a href="#">Security</a> |                            |
| <input type="checkbox"/> |  | GeneratedViews | <a href="#">Attributes</a> | <a href="#">Security</a> | <a href="#">Categories</a> |
| <input type="checkbox"/> |  | Interconnects  | <a href="#">Attributes</a> | <a href="#">Security</a> | <a href="#">Categories</a> |
| <input type="checkbox"/> |  | Metadata       | <a href="#">Attributes</a> | <a href="#">Security</a> | <a href="#">Categories</a> |
| <input type="checkbox"/> |  | Shares         | <a href="#">Attributes</a> | <a href="#">Security</a> | <a href="#">Categories</a> |
| <input type="checkbox"/> |  | System         | <a href="#">Attributes</a> | <a href="#">Security</a> | <a href="#">Categories</a> |
| <input type="checkbox"/> |  | WSDLs          | <a href="#">Attributes</a> | <a href="#">Security</a> | <a href="#">Categories</a> |
| <input type="checkbox"/> |  | Categories     | <a href="#">Attributes</a> | <a href="#">Security</a> | <a href="#">Categories</a> |

- Step 2** In the Browse Directories area, click on directory names to navigate to the directory in which the share appears.

**Step 3** Click **Add File to Share**. The Upload File screen appears.

**Upload File**

To upload a file to the grid directory and to the machine on which the share resides, specify the file name below or click **Browse...** and navigate to the directory that contains the file, then select the file.

File name:   (Overwrite existing file? )

**Step 4** Specify the file name or click **Browse...** and navigate to the local directory that contains the file, then select the file. If a file of that name already exists in this share and you want to overwrite it, click the overwrite existing file box.

**Step 5** Click **Submit** to upload the file.

---

**Note** After you add a file to an Avaki share, we recommend that you examine and if necessary reset the file's permissions. For instructions on viewing and modifying permissions, see [“Using access control lists” on page 237](#).

---

## Moving shared data

You can move the source directory of an Avaki share to another location in your local file system without recreating the Avaki share from scratch. The new location must be on a machine that is a member of the same grid domain.

If you simply move your files and directories and create another Avaki share pointing to the new location, you must redo any configuration changes you have made to the share. Following the procedure given here lets you preserve configuration changes you have made to the share. Configuration changes include:

- Changes to the ACLs of the share or its contents
- Changes to the attributes of the share or its contents
- Addition of share servers to the share
- Changes to the share's refresh schedule
- Changes to the LBF values of the share's share servers
- Changes to the share's encryption level

Follow these steps to move an Avaki share's source directory.

- Step 1** Following the instructions in the procedure [“Taking Avaki shares off-line” on page 285](#), take the Avaki share whose data you plan to move off-line.
- Step 2** Move the data to its new location. Make a note of the new path.
- Step 3** Navigate to the View Avaki Shares screen:

Home > Service management > View Avaki shares

| View Avaki Shares                                                                                                                                                                                                                                                    |                              |                           |                          |                            |                            |                          |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------|---------------------------|--------------------------|----------------------------|----------------------------|--------------------------|
| The following Avaki shares have been integrated into the local grid domain. To shut down a share, click <b>Disconnect</b> . The share will be removed from the grid directory system, but the corresponding files in the underlying file system will not be deleted. |                              |                           |                          |                            |                            |                          |
| Avaki Share Name                                                                                                                                                                                                                                                     | Creation Date                | View/Edit Share           | Schedule                 | Shut Down                  | Attributes                 | Security                 |
| Bam-bamsfiles                                                                                                                                                                                                                                                        | Tue Sep 21 23:40:53 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Barneysfiles                                                                                                                                                                                                                                                         | Tue Sep 21 23:37:33 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Bettysfiles                                                                                                                                                                                                                                                          | Tue Sep 21 23:38:07 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="button" value="Done"/>                                                                                                                                                                                                                                  |                              |                           |                          |                            |                            |                          |

- Step 4** Click **View/Edit** beside the Avaki share you want to modify. The Update Avaki Share Information window appears.

| Update Avaki Share Information                                                                                                                                                                                   |                                                                        |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------|
| You are viewing information about the selected share. To update the information, enter the new values and click <b>Submit</b> . Note that you may not update the share's local path unless the share is offline. |                                                                        |
| <b>Share name:</b>                                                                                                                                                                                               | Barneysfiles                                                           |
| <b>Creation date:</b>                                                                                                                                                                                            | Tue Sep 21 23:37:33 EDT 2004                                           |
| <b>Grid server:</b>                                                                                                                                                                                              | localhost:1099                                                         |
| <b>Grid path:</b>                                                                                                                                                                                                | /Shares/Barneysfiles                                                   |
| <b>Local path:</b>                                                                                                                                                                                               | C:\Documents and Settings\bthoene                                      |
| <b>Encryption level:</b>                                                                                                                                                                                         | <input checked="" type="radio"/> Clear <input type="radio"/> Encrypted |
| <b>Status:</b>                                                                                                                                                                                                   | <input checked="" type="radio"/> Online <input type="radio"/> Offline  |
| <b>Description:</b>                                                                                                                                                                                              | <input type="text"/>                                                   |
| <input type="button" value="Submit"/> <input type="button" value="Cancel"/>                                                                                                                                      |                                                                        |

- Step 5** In the Local path field, enter the new path to the source directory of the share.



- Step 6** In the Status field, click the **Online** button.
- Step 7** Click **Submit** to bring the share on line at its new location.

## Taking Avaki shares on- and off-line

The procedures in this section tell you how to temporarily remove an Avaki share from the data catalog and how to restore it again. These procedures have no effect on the shares' source files and directories; they change only what is visible and accessible through the data catalog.

If you want to permanently disconnect an Avaki share, see [“Disconnecting Avaki shares” on page 288](#).

### Taking Avaki shares off-line

Follow these steps to take an Avaki share temporarily off-line.

- Step 1** Log in to the Avaki domain as a user who has write permission on the Avaki share that you want to take off-line.
- Step 2** Navigate to the View Avaki Shares screen:  
Home > Service management > View Avaki shares

**View Avaki Shares**

The following Avaki shares have been integrated into the local grid domain. To shut down a share, click **Disconnect**. The share will be removed from the grid directory system, but the corresponding files in the underlying file system will not be deleted.

| Avaki Share Name | Creation Date                | View/Edit Share           | Schedule                 | Shut Down                  | Attributes                 | Security                 |
|------------------|------------------------------|---------------------------|--------------------------|----------------------------|----------------------------|--------------------------|
| Bam-bamsfiles    | Tue Sep 21 23:40:53 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Barneysfiles     | Tue Sep 21 23:37:33 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Bettysfiles      | Tue Sep 21 23:38:07 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

- Step 3** Click **View/Edit** beside the Avaki share you want to modify. The Update Avaki Share Information window appears.

**Update Avaki Share Information**

You are viewing information about the selected share. To update the information, enter the new values and click **Submit**. Note that you may not update the share's local path unless the share is offline.

**Share name:** Barneysfiles  
**Creation date:** Tue Sep 21 23:37:33 EDT 2004  
**Grid server:** localhost:1099  
**Grid path:** /Shares/Barneysfiles  
**Local path:** C:\Documents and Settings\bthoene

**Encryption level:**  Clear  Encrypted  
**Status:**  Online  Offline

**Description:**

- Step 4** In the Status field, choose **Offline**.

- Step 5** Click **Submit** to take the share off-line. The system displays a confirmation page.

## Bringing Avaki shares on-line

Follow these steps to restore an Avaki share to service after it has been taken off-line.

- Step 1** Log in to the Avaki domain as a user who has write permission on the Avaki share that you want to bring on-line.
- Step 2** Navigate to the View Avaki Shares screen:

Home > Service management > View Avaki shares

### View Avaki Shares

The following Avaki shares have been integrated into the local grid domain. To shut down a share, click **Disconnect**. The share will be removed from the grid directory system, but the corresponding files in the underlying file system will not be deleted.

| Avaki Share Name | Creation Date                | View/Edit Share           | Schedule                 | Shut Down                  | Attributes                 | Security                 |
|------------------|------------------------------|---------------------------|--------------------------|----------------------------|----------------------------|--------------------------|
| Barn-barnsfiles  | Tue Sep 21 23:40:53 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Barneysfiles     | Tue Sep 21 23:37:33 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Bettysfiles      | Tue Sep 21 23:38:07 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

- Step 3** Click **View/Edit** beside the Avaki share you want to modify. The Update Avaki Share Information window appears.

### Update Avaki Share Information

You are viewing information about the selected share. To update the information, enter the new values and click **Submit**. Note that you may not update the share's local path unless the share is offline.

**Share name:** Barneysfiles

**Creation date:** Tue Sep 21 23:37:33 EDT 2004

**Grid server:** localhost:1099

**Grid path:** /Shares/Barneysfiles

**Local path:** C:\Documents and Settings\bthoene

**Encryption level:**  Clear  Encrypted

**Status:**  Online  Offline

**Description:**

- Step 4** In the Status field, choose **Online**.

- Step 5** Click **Submit** to bring the share on-line. The system redisplay the Update Avaki Share Information window with the status set to online.

## Disconnecting Avaki shares

Follow these steps to shut down an Avaki share. This procedure removes the shared files and directories from the data catalog, but leaves the source files and directories intact.

If you change your mind after you've disconnected an Avaki share, you must recreate the share to get it back, and any configuration changes you have made to the share will be lost. (Configuration changes include changes to the ACLs or attributes of the share or its contents; addition of share servers; and changes to LBF values, the encryption level, and the refresh schedule.)

If you want to temporarily disconnect an Avaki share, see [“Taking Avaki shares off-line” on page 285](#).

- Step 1** Navigate to the View Avaki Shares screen:

Home > Service management > View Avaki shares

**View Avaki Shares**

The following Avaki shares have been integrated into the local grid domain. To shut down a share, click **Disconnect**. The share will be removed from the grid directory system, but the corresponding files in the underlying file system will not be deleted.

| Avaki Share Name | Creation Date                | View/Edit Share           | Schedule                 | Shut Down                  | Attributes                 | Security                 |
|------------------|------------------------------|---------------------------|--------------------------|----------------------------|----------------------------|--------------------------|
| Bam-bamsfiles    | Tue Sep 21 23:40:53 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Barneysfiles     | Tue Sep 21 23:37:33 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| Bettysfiles      | Tue Sep 21 23:38:07 EDT 2004 | <a href="#">View/Edit</a> | <a href="#">Schedule</a> | <a href="#">Disconnect</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

- Step 2** In the Shut Down column, click **Disconnect** for the share you want to shut down.
- Step 3** A confirmation window appears; click **OK** to disconnect the share. The system redisplay the View Avaki Shares screen without the share that you disconnected.

# *Managing interconnections*

---

This chapter explains how to make data in one Avaki domain accessible to users in another domain.

---

## ***About interconnecting Avaki domains***

An *interconnection* is a unidirectional link between two Avaki domains. One domain is a data provider, and the other is a data consumer—users in the consumer domain acquire access to data in the provider domain. To access data in another Avaki domain,

- The consumer domain creates a connection to the provider domain.
- The consumer domain gives the provider information about a user or group who needs to access the data in the provider domain.
- The provider domain gives that user or group permission to access the files or directories.

An interconnection creates links in the consumer domain's data catalog to the top level of the provider domain's data catalog. These links enable the consumer domain to send requests to specific objects in the provider domain.

You can create interconnections in both directions to enable users in each Avaki domain to access data in the other, making each domain both a provider and a consumer.

## Proxy servers and proxy routing tables

A proxy server enables traffic to pass from a consumer domain to a provider domain that is not directly reachable. The proxy server, which is part of the provider domain, transparently and passively forwards requests into its Avaki domain and forwards the replies back out to the consumer domain. Because proxy servers operate transparently, interconnections work the same way whether a proxy server is involved or not.

A proxy routing table is part of a consumer domain. It tells the consumer domain how to route requests for data in provider domains via the providers' proxy servers.

When a request to an Avaki object is initiated, two questions arise:

- Is this object in another Avaki domain?
- If so, is there a proxy routing table entry for this request's destination?

Suppose a user in a consumer domain requests a file in an interconnected provider domain. If there is an entry for the provider domain in the consumer's proxy routing table, the request is routed through the provider's proxy server instead of directly to the destination. The proxy server forwards the request to the destination object in the provider domain, and returns the provider object's reply back to the requestor.

## The provider domain

This section provides an overview of the interconnection tasks required on the provider side.

### Is a proxy server needed?

As the administrator of a provider domain who wants to offer data access to users in a consumer domain, your first task is to determine whether you need to set up a proxy server to support the interconnection. You must set up a proxy server unless the two Avaki domains share a uniform IP space—that is, if you can ping from one to the other, you don't need a proxy server. If the two domains are separated by a firewall or Network Address Translation (NAT), they don't share uniform IP space and you do need a proxy server.

### Provider tasks

When you interconnect a provider domain to a consumer domain, you'll complete the following tasks:

1. Set up a proxy server if one is required (see [“Is a proxy server needed?”](#) above). For set-up instructions, see [“Enabling interconnection access” on page 299](#).

2. If you set up a proxy server, send information about it to the administrator of the consumer domain. (Details on what to send appear in the proxy server set-up procedure.) This information will be used to configure the consumer domain's proxy routing table.
3. Enable users or groups from the consumer domain to access only the data you explicitly allow them to see. For more information, see [“User access methods” on page 295](#).
4. If you want to receive update notifications from views in the other domain, see [“Enabling cross-domain messaging” on page 312](#).

## The consumer domain

This section provides an overview of the interconnection tasks required on the consumer side.

1. If the provider domain has a proxy server, enter details about the proxy server into your domain's proxy routing table. See [“Configuring proxy routing tables” on page 292](#). (Skip this step if the provider domain does not have a proxy server.)
2. Create the link to the provider domain as described in [“Setting up the interconnection” on page 293](#).
3. If the interconnection between the two domains will go both ways, decide which of two methods you'll use to give your users access to files or directories in the provider domain. See the section [“User access methods” on page 295](#).
4. Execute the user access method you've chosen.
5. If you want to receive update notifications from views in the other domain, see [“Enabling cross-domain messaging” on page 312](#).

## Other tasks

This chapter also covers the following topics:

- [“Viewing interconnected domains” on page 311](#)
- [“Disconnecting interconnected domains” on page 315](#)

---

## Connecting to a provider domain

To connect your Avaki domain to a provider domain, you must complete the following tasks:

- [“Configuring proxy routing tables,”](#) below. (Skip this task if the provider domain does not have a proxy server.)
- [“Setting up the interconnection” on page 293.](#)
- [“User access methods” on page 295.](#) (If the interconnection is one way, you must use the interconnect ID method described in that section.)
- Execute the user access method you’ve chosen. The section [“User access methods” on page 295](#) refers you to the appropriate procedures for each method.

### Configuring proxy routing tables

If the data in the provider domain is accessed through a proxy server, you must configure your domain to communicate with the provider domain’s proxy server. Do the following:

**Step 1** From the administrator of the provider domain, get the following information:

- The name of the provider domain
- The proxy server’s IP address or DNS name. We recommend that you use whichever identifier will be more stable. (For example, if the proxy server operates in a DHCP environment where IP addresses change, but host names are constant, use the host name.)
- The listening port number on the proxy server (default ports are 18443 for SSL communication (HTTPS) or 18080 for non-SSL communication (HTTP)).
- Whether the listening port is SSL or non-SSL.

**Step 2** Navigate to the Proxy Routing Table screen:



## Home &gt; Domain interconnection management &gt; Configure proxy routing table

| Proxy Routing Table                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                      |                                       |                                                                  |                                                                  |              |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------|---------------------------------------|------------------------------------------------------------------|------------------------------------------------------------------|--------------|
| Grid Domain                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Proxy IP/Host Name   | Proxy Port                            | SSL Port?                                                        | Use Data Compression?                                            | Delete Entry |
| <p><b>Add Proxy Entry</b></p> <p>If your domain needs to access data on a remote domain via a proxy server, you must configure your domain to communicate with the remote domain's proxy server as follows:</p> <ol style="list-style-type: none"> <li>To add a proxy server to the routing table, specify the grid domain, proxy server IP address/host name, and proxy server port (some common port values are 18443 for SSL communication or 18080 for non-SSL communication). Select whether this port is an SSL or non-SSL port. Select whether compression should be used for communication with this proxy, then click <b>Submit</b>.</li> <li>Go to the <a href="#">Create Interconnection</a> screen to create an interconnection from your grid domain to the remote domain.</li> </ol> |                      |                                       |                                                                  |                                                                  |              |
| Grid Domain                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Proxy IP/Host Name   | Proxy Port                            | SSL Port?                                                        | Use Data Compression?                                            |              |
| <input type="text"/>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | <input type="text"/> | <input type="text"/>                  | <input checked="" type="radio"/> Yes<br><input type="radio"/> No | <input checked="" type="radio"/> Yes<br><input type="radio"/> No |              |
| <input type="button" value="Submit"/>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |                      | <input type="button" value="Cancel"/> |                                                                  |                                                                  |              |

- Step 3** To add a proxy server to the routing table, enter the information you obtained in Step 1. Select whether compression should be used for communication via this proxy.
- Step 4** Click **Submit** to add the entry to the proxy routing table.

## Setting up the interconnection

- Step 1** Obtain the following information from the administrator of the provider domain:
- The name of the provider domain
  - The provider GDC's host name or IP address
  - The provider GDC's connect port number

This information is required for all interconnections, whether a proxy server is involved or not.

- Step 2** To establish a connection with the provider domain, navigate to the Create Interconnection screen:

## Home &gt; Domain interconnection management &gt; Create interconnection

**Create Interconnection**

To access data in a connected grid domain, do the following:

1. If the data on the provider domain is accessed through a proxy server, configure your domain to communicate with the provider domain's proxy server. Go to the [Configure Proxy Routing Table](#) screen and add the provider domain's proxy server to the routing table.
2. To establish a connection with the provider domain, specify the domain's connection information below. For "Interconnection description," provide a brief description of the interconnection, such as whether it is internal or intercompany and why it was created.
3. Click **Submit** to connect the active domain on **FRANCIUM** to the active domain on the server specified below.
4. Go to the [Select User](#) or [Select Group](#) screen. Click the **View/Edit User** or **View/Edit Group** button beside the name of the user or group that should have access to the provider domain. On the screen that appears, Click the **Download Interconnect ID** link. Save the interconnect ID to your local machine, then send it by email to the provider domain's administrator.
5. To enable your domain to access a directory in the provider domain, the provider domain's administrator must configure a proxy server, upload an interconnect ID from your domain, and modify the directory's permissions. For more information, see the [Avaki Administration Guide](#).
6. To enable your domain to receive messages from the provider domain, make sure the provider domain has established an interconnection to your domain. Then follow the **Configure Messaging** link for the provider domain on the [View Interconnections](#) page.

|                                                   |                                   |
|---------------------------------------------------|-----------------------------------|
| Name of grid domain to connect with:              | <input type="text"/>              |
| DNS name or IP address of server to connect with: | <input type="text"/>              |
| Connect port of server to connect with:           | <input type="text" value="1099"/> |
| Interconnection description:                      | <input type="text"/>              |

**Step 3** Fill in the form:

- Enter the name of the Avaki domain to connect with.
- Enter the DNS name or IP address of the provider domain's GDC.
- Enter the GDC's server connect port number (defaults to 3099). The connect port is the port the GDC uses to communicate with other objects in the data grid.
- Enter a brief description of the interconnection, such as whether it is internal or intercompany and why it was created.

**Step 4** Click **Submit** to connect the active domain to the active domain on the specified server.

## User access methods

There are two methods for giving users in a consumer domain access to files and directories in a provider domain:

- Sending the user's or group's interconnect IDs to an administrator of the provider domain
- Exposing the users to an administrator of the provider domain (for two-way interconnects only)

**Sending interconnect IDs to the provider administrator.** This method requires you to send the user's or group's interconnection ID to the administrator of the provider domain; it's described in the section [“Submitting user or group interconnection IDs,”](#) below. The provider adds the interconnection ID to the directory's or file's ACL, as described in the section [“Enabling user or group access” on page 305](#). You must use this method if your interconnection with the provider domain is one-way. You should also use this method if you don't trust the provider administrator enough to give him or her access to your domain's user accounts.

**Exposing users to the provider administrator.** If you have a two-way interconnect between two grid domains, you can use another method to enable users in your domain to access grid directories or files in another grid domain. This method requires you to expose your users to a trusted administrator of the provider domain, who then has write permission on those user accounts. The provider administrator can then use the security UI to add the users to ACLs of objects in his or her domain without requiring interconnect IDs for those users. The users can even be added to groups in the provider domain. When you expose users, the provider administrator is added to the ACLs of the users being exposed, which means the provider administrator can edit the user profiles in other ways, though the provider administrator cannot delete users exposed from your domain. This method of providing access is described in the section [“Exposing users in a two-way interconnection” on page 309](#).

## Submitting user or group interconnection IDs

This section explains how to generate and send user or group interconnection IDs to a provider domain—the first step in the interconnect ID method of enabling interdomain access. This section covers the following topics:

- [“Sending a user interconnection ID” on page 296](#)
- [“Sending a group interconnection ID” on page 298](#)

## Sending a user interconnection ID

To send a user interconnection ID to a provider domain, do the following:

**Step 1** Navigate to the Select User screen:

Home > User management > View and modify users

### Select User

To modify a user's personal information, download a user interconnect ID, or change a password, select the user and click the **View/Edit** button. To delete a user from a grid domain, select the user and click **Delete All Checked**. To configure the user ID (UID) that is mapped to a user account, select the user and click **Update UIDs** below.

If you have a two-way connection with a remote domain, you may want to enable the administrator of the remote domain to view users in your domain and add them to groups in the remote domain. To do this, you must give the administrator permission to view and edit those users, as follows:

1. Obtain the interconnect ID that represents the administrator of the remote domain.
2. In the table below, select the users whose settings you want to allow the remote administrator to modify.
3. Go to the [Expose Users](#) screen to continue.

**Note:** The remote administrator will be able to edit only the users you selected, but the administrator will be able to view *all* the users in your domain.

All users

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

| Select                   | Username      | Authentication Service | View/Edit User            | Attributes                 | Security                 |
|--------------------------|---------------|------------------------|---------------------------|----------------------------|--------------------------|
| <input type="checkbox"/> | Administrator | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | MessagingUser | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | cmagadieu     | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

**Total number of users in domain: 3**

- Step 2** Click the **View/Edit User** button beside the name of the user who should have access to the provider domain. The View/Modify User Info screen appears.

**View/Modify User Info**

Username: cmagadieu

Full name:

Email:

UID for NFS: unknown

Groups:

| Group Name  | Domain       |
|-------------|--------------|
| DomainUsers | CherylDomain |

**Download User Interconnect ID**

To send the user's interconnection ID to a remote domain, click **Download Interconnect ID**.

[Download Interconnect ID](#)

- Step 3** Click the **Download Interconnect ID** link. Save the interconnect ID text file to your local machine.

- Step 4** Send the interconnect ID file by email to the provider domain's administrator.

The procedure the provider administrator must follow to use the interconnect ID appears in the section [“Enabling user or group access” on page 305](#).

## Sending a group interconnection ID

To send a group interconnection ID to a provider domain, do the following:

**Step 1** Navigate to the Select Group screen:

Home > User management > View and modify groups

**Select Group**

All groups

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

To modify a group's description, download a group interconnect ID, or add or remove group members, select the group and click the **View/Edit** button. To delete a group from a grid domain, select the group and click **Delete All Checked**. To configure the group ID (GID) that is mapped to a group account, select the group and click **Update GIDs** below.

| Select                   | Group name             | Authentication service | View/Edit Group           | Attributes                 | Security                 |
|--------------------------|------------------------|------------------------|---------------------------|----------------------------|--------------------------|
| <input type="checkbox"/> | Administrators         | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | DataProviders          | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | DatabaseAdministrators | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | DomainUsers            | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | MessagingUsers         | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | UserAdministrators     | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | testusers              | myNis                  | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | users                  | myNis                  | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

- Step 2** Click the **View/Edit Group** button beside the name of the group whose members should have access to the provider domain. The Modify Group Description screen appears.

**Modify Group Description**

Group name: testusers  
 GID for NFS: 104  
 Description:

**Download Group Interconnect ID**

To send the group's interconnection ID to a remote domain, click **Download Interconnect ID**.

[Download Interconnect ID](#)

- Step 3** Click the **Download Interconnect ID** link. Save the interconnect ID text file to your local machine.

- Step 4** Send the interconnect ID file by email to the provider domain's administrator.

The procedure the provider administrator must follow to use the interconnect ID appears in the section [“Enabling user or group access” on page 305](#).

---

## ***Enabling interconnection access***

To enable a consumer domain to access a directory or select a data object in your grid domain, you must do the following:

1. If you are interconnecting domains that are on separate networks, configure your domain to allow a consumer domain to access it via a proxy server. For instructions, see [“Configuring a proxy server,”](#) below.
2. Enable users or groups from the consumer domain to access a directory or select a data object in your grid domain. If this is a one-way interconnect, see [“Enabling user or group access” on page 305](#). If this is a two-way interconnect, you can also use that method, or you can use the method described in [“Exposing users in a two-way interconnection” on page 309](#); for help choosing, see [“User access methods” on page 295](#).

## Configuring a proxy server

### Setting the host name

(Optional) Specify a host name or IP address for your proxy server to advertise if the default (the server machine's name) is unsuitable. For instructions, see [“Setting a server's host name or IP address” on page 32](#).

### Using nondefault ports

If the ports that proxy servers use by default conflict with ports already in use on the proxy server machine, you can specify other ports for your proxy server to use. (For a list of default ports, see [“Planning use of network ports” on page 8](#).) Follow these steps to change the default ports.

---

**Note** Do not reassign ports on an Avaki server once it's been joined to an Avaki domain; doing so will cause communication problems between servers. If the server has been joined to a domain, you must re-install and reconfigure the server, change the ports, and join the new server to the Avaki domain.

- Step 1** In a text editor, open `<Avaki-install-dir>/jboss/server/proxy-server/conf/bindings.xml`. Two ports are specified in this file: 18443 and 18080. If necessary, replace these port numbers with values appropriate for this machine. Save the file.
- Step 2** Record the port numbers you enter in place of 18443 (the HTTPS listening port) and 18080 (the HTTP listening port). In the procedure that follows, “Proxy server configuration,” you'll configure your firewall to forward these ports.
- Step 3** In a text editor, open `<Avaki-install-dir>/jboss/server/proxy-server/conf/join.properties`. Two ports are specified in this file: 1199 and 38080. If necessary, replace these port numbers with values appropriate for this machine.
- Step 4** Record the port number you enter in place of 1199 (the first port number in the file); this is the proxy server's connect port, and you must supply the connect port number when you connect this proxy server to the Avaki domain.

### Proxy server configuration

To configure your machine to allow a consumer domain to access it via a proxy server, do the following:

- Step 1** Configure your firewall to forward the Avaki proxy server's HTTPS listening port (TCP port 18443, by default) to the proxy server's IP address. If you want to allow



non-SSL access, you must also configure your firewall to forward the proxy server's HTTP listening port (TCP port 18080, by default) to the proxy server's IP address.

- Step 2** Load Avaki software and run the Avaki installer on the machine on which the proxy server will run. The proxy machine must be a machine that can have a port forwarded to it from the firewall. Choose a Firewall Proxy Server installation. (See [Chapter 2](#) for more on installing the software.)
- Step 3** If you have chosen to generate self-signed SSL certificates for your Avaki servers, set up this proxy server to do so. For instructions, see [“Generating self-signed SSL certificates” on page 34](#).
- Step 4** You can start the proxy server with or without the auto-restart option enabled. When auto-restart is enabled, the server restarts automatically when the computer restarts. To initialize the proxy server, do the following:

**In Windows**, Do either of the following to initialize the proxy server with auto-restart enabled:

- Open the Start menu. In the Avaki Data Grid program group, select **Proxy Server**, then select the **Register ProxyServer as Windows Service** shortcut.
- In the Avaki installation directory (which defaults to <system\_drive\_root>\AvakiDataGrid<release-number> (for example, C:\AvakiDataGrid70)), enter this command:

```
C:\> proxy-server --register
```

The server prompts you for:

- A user name. Enter the name of the nonadministrative user account with logon-as-service privileges that you set up to run Avaki services (see [“Configuring user accounts” on page 11](#)). Enter the Windows domain name before the user name, for example, BEDROCK\avaki.
- A password. Enter the password for the user you entered above.

The server takes a minute or two to start. When it's done, you can find an entry for the proxy server in the machine's Services list (Start > Control Panel > Administrative Tools > Services).

---

**Note** Use the Windows Services list to stop and restart Avaki servers that are registered as services. Right-click on the server name (for example, Avaki-Proxy-Service-1096608202) and select **Stop** or **Restart** from the menu.

To start the proxy server without enabling auto-restart, change to the Avaki installation directory (by default, <system\_drive\_root>\AvakiDataGrid<release-number> (for example, C:\AvakiDataGrid70)) and issue a command of this form:

```
proxy-server --start [--user=<user>]
```

where <user> is the nonadministrative user account that you set up to run Avaki services (for more information, see [“Configuring user accounts” on page 11](#)).

For example:

```
$ proxy-server --start --user=avaki
```

The server prompts you to enter a password for the user account you specify with the --user option.

The server is started when a message similar to the following appears:

```
12:13:50,659 INFO [Server] JBoss (MX MicroKernel) [3.0.6
(CVSTag=JBoss_3_0_6 Date=200301260037)] Started in
0m:47s:988ms
```

**In Unix:** The proxy server startup script is in the Avaki installation directory (such as /root/AvakiDataGrid70). To start the proxy server with auto-restart enabled, log in as root, switch to your Avaki installation directory and issue a command of this form:

```
proxy-server --register [--user=<name>]
```

To start the proxy server without enabling auto-restart, switch to your Avaki installation directory and issue the following command:

```
$ proxy-server --start
```

The server is started when a message similar to the following appears:

```
2005-07-29 16:53:46,669 INFO
[org.jboss.system.server.Server, main] JBoss (MX
MicroKernel) [3.0.6 (CVSTag=JBoss_3_0_6 Date=200301260037)]
Started in 0m:33s:848ms
```

**Step 5** To connect a proxy server to the grid, open a web browser and point it to

```
http://<gdc-name>:7080
```

where <gdc-name> is the name of the grid domain controller to which you're joining the new proxy server.

**Note** When you connect a new server to an existing Avaki domain, the connection is made *from* the GDC *to* the server—so make sure your browser is connected to the GDC and not to the proxy server machine.

- Step 6** Install an SSL certificate on the proxy server. For instructions, see [“Installing an SSL certificate” on page 39](#).
- Step 7** Log in to Avaki as a member of the Administrators group and navigate to the Connect Proxy Server screen:

Home > Server management > Connect new proxy server

**Connect Proxy Server**

To connect a proxy server to the grid, install the proxy server on the machine where the server will reside, start the proxy server, and then fill in the information below, specify the server's logical name, machine name, and connect port.

Logical name for new proxy server:

IP or DNS name of new proxy server:

Connect port of new proxy server:

- Step 8** Fill in the form:
- Logical name for new proxy server: Enter a name for the proxy server.
  - IP or DNS name of new proxy server: Enter the fully qualified DNS name or IP address of the machine on which the proxy server is running. (A fully qualified DNS name is `bedrock.avaki.com`, for example, rather than just `bedrock`.)
  - Connect port of new proxy server: Enter the server connect port number (the default is 1199) for this proxy server. (You need to enter a connect port number only if you started the proxy server with a connect port other than the default, 1199.)

**Note** You can find an Avaki server's connect port (as well as its name) in its `connectinfo.txt` file, which is generated when the server starts. The path to the file is:

<Avaki-install-dir>/jboss/server/proxy-server/log/connectinfo.txt

(<Avaki-install-dir> is the local directory where Avaki software is installed.)

**Step 9** Click **Submit** to set up the proxy server.

**Step 10** Give the following information to the administrator of the consumer domain:

- The name of your Avaki domain
- Your proxy server's IP address or host name. We recommend that you use whichever identifier will be more stable. (For example, if the proxy server operates in a DHCP environment where IP addresses change, but host names are constant, use the host name.) For information on setting a name for the proxy server to advertise, see [“Setting a server's host name or IP address” on page 131](#).
- The number of your proxy server's listening port (default ports are 18443 for SSL communication (HTTPS) or 18080 for non-SSL communication (HTTP)). Be sure to specify SSL or non-SSL.
- Your GDC's host name or IP address and connect port number

This information is used to configure the consumer domain's proxy routing table and to set up the interconnection.

## Enabling user or group access

If you have received a user or group interconnect ID from a consumer domain, you can enable the user or group to access a directory or a data object in your Avaki domain by adding the interconnect ID to the access control list (ACL) of the directory or object. Do the following:









- Step 1** Save the interconnect ID file to a local machine.
- Step 2** Navigate to the Browse Directories screen:

Home > Data catalog management > Browse directories

### Browse Directories

The top-level directory in a grid domain is /. Underneath is the /System subdirectory, which contains directories and files related to a particular grid domain, and the /Interconnects link to the Interconnects directory, which contains the root directories of any domains to which a domain is interconnected. A grid domain's administrator can view and modify the settings for an interconnected domain if the administrator has been granted access rights for modifying the domain.

You can add an Avaki directory at any level if you have write permission for the directory in which you want to create a directory. By default, only the Avaki administrator can create new directories at the top level, /.

|                          |                                                                                     |                |                            |                                                     |
|--------------------------|-------------------------------------------------------------------------------------|----------------|----------------------------|-----------------------------------------------------|
| <input type="checkbox"/> |    | /              | <a href="#">Attributes</a> | <a href="#">Security</a>                            |
| <input type="checkbox"/> |    | GeneratedViews | <a href="#">Attributes</a> | <a href="#">Security</a> <a href="#">Categories</a> |
| <input type="checkbox"/> |    | Interconnects  | <a href="#">Attributes</a> | <a href="#">Security</a> <a href="#">Categories</a> |
| <input type="checkbox"/> |    | Metadata       | <a href="#">Attributes</a> | <a href="#">Security</a> <a href="#">Categories</a> |
| <input type="checkbox"/> |    | Shares         | <a href="#">Attributes</a> | <a href="#">Security</a> <a href="#">Categories</a> |
| <input type="checkbox"/> |  | System         | <a href="#">Attributes</a> | <a href="#">Security</a> <a href="#">Categories</a> |
| <input type="checkbox"/> |  | WSDLs          | <a href="#">Attributes</a> | <a href="#">Security</a> <a href="#">Categories</a> |
| <input type="checkbox"/> |  | Categories     | <a href="#">Attributes</a> | <a href="#">Security</a> <a href="#">Categories</a> |

- Step 3** On the Browse Directories screen, click on the directory names to navigate to the object or directory for which access will be allowed.

**Step 4** Click the **Security** link. The View Security Information screen appears.

### View Security Information

You are viewing security information for the following object:  
**/System/Domains/Bedrock/Services/ShareManagers/Water Buffalo Lodge\_1130096305646**

The current owner of the object is the User **Administrator** in domain **Bedrock** and the Grid authentication service **DefaultAuthService**.

The following users and groups have been added to the Access Control List (ACL) for this object. To modify a user or group's permissions, place a check mark next to the user or group and click **Edit All Checked**. To add a user who is in the current grid domain, click **Add User to ACL**. To add a group that is in the current domain, click **Add Group to ACL**. To add a user or group that is in a connected domain, click **Add Via Interconnect ID**.

| Select                   | Name          | Type | Domain  | Auth Service       | Auth Service Type | Read  | Write | Execute | Delete |
|--------------------------|---------------|------|---------|--------------------|-------------------|-------|-------|---------|--------|
| <input type="checkbox"/> | Administrator | User | Bedrock | DefaultAuthService | Grid              | Allow | Allow | Allow   | Allow  |
| <input type="checkbox"/> | wilma         | User | Bedrock | DefaultAuthService | Grid              | Allow | Allow | Allow   | Allow  |

#### Manage Access Control Lists

[Edit All Checked](#)      Configure permissions for each user or group in the ACL for an object  
[Add User to ACL](#)      Add a user to the ACL for an object  
[Add Group to ACL](#)      Add a group to the ACL for an object  
[Add Via Interconnect ID](#)      Upload an interconnect ID to add a user or group to the ACL for an object

**Step 5** Click the **Add Via Interconnect ID** link. The Add User or Group Via Interconnect ID screen appears.

### Add User or Group Via Interconnect ID

To upload an interconnect ID, click **Browse** and navigate to the directory where you stored the user or group interconnect ID. Select the interconnect ID, then click **Submit** to add the user or group to the access control list (ACL) for the following object:

**/Shares/MyAvakiShare**

File name:

- Step 6** Click the **Browse** button and navigate to the directory on your local machine where you stored the user or group interconnect ID. Select the interconnect ID, then click **Upload** to add the user or group to the ACL for the directory. The Modify Permissions screen appears.

**Modify Permissions**

You may specify read, write, execute, or delete permission for each user or group in the ACL for the following object:

**/System/Domains/Bedrock/Services/ShareManagers/Water Buffalo Lodge\_1130096305646**

For each permission, you may specify the allow, deny, or unset option, or you may leave the current value as is. (If you choose the unset option for all permissions and the user is not the object's owner, the user will be removed from the ACL.)

Select the new permissions for the user or group:

| Name           | Type    | Domain  | Auth Service       | Auth Service Type | Set as Owner             | Delete                   | Permissions                                                                                                                                                                                                                                                                                                                                                                                                                            |  |         |     |             |       |         |              |       |         |                |       |         |               |       |         |
|----------------|---------|---------|--------------------|-------------------|--------------------------|--------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|---------|-----|-------------|-------|---------|--------------|-------|---------|----------------|-------|---------|---------------|-------|---------|
| wilma          | User    | Bedrock | DefaultAuthService | Grid              | <input type="checkbox"/> | <input type="checkbox"/> | <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th></th> <th>Current</th> <th>New</th> </tr> </thead> <tbody> <tr> <td><b>Read</b></td> <td>Allow</td> <td>As is ▾</td> </tr> <tr> <td><b>Write</b></td> <td>Allow</td> <td>As is ▾</td> </tr> <tr> <td><b>Execute</b></td> <td>Allow</td> <td>As is ▾</td> </tr> <tr> <td><b>Delete</b></td> <td>Allow</td> <td>As is ▾</td> </tr> </tbody> </table> |  | Current | New | <b>Read</b> | Allow | As is ▾ | <b>Write</b> | Allow | As is ▾ | <b>Execute</b> | Allow | As is ▾ | <b>Delete</b> | Allow | As is ▾ |
|                | Current | New     |                    |                   |                          |                          |                                                                                                                                                                                                                                                                                                                                                                                                                                        |  |         |     |             |       |         |              |       |         |                |       |         |               |       |         |
| <b>Read</b>    | Allow   | As is ▾ |                    |                   |                          |                          |                                                                                                                                                                                                                                                                                                                                                                                                                                        |  |         |     |             |       |         |              |       |         |                |       |         |               |       |         |
| <b>Write</b>   | Allow   | As is ▾ |                    |                   |                          |                          |                                                                                                                                                                                                                                                                                                                                                                                                                                        |  |         |     |             |       |         |              |       |         |                |       |         |               |       |         |
| <b>Execute</b> | Allow   | As is ▾ |                    |                   |                          |                          |                                                                                                                                                                                                                                                                                                                                                                                                                                        |  |         |     |             |       |         |              |       |         |                |       |         |               |       |         |
| <b>Delete</b>  | Allow   | As is ▾ |                    |                   |                          |                          |                                                                                                                                                                                                                                                                                                                                                                                                                                        |  |         |     |             |       |         |              |       |         |                |       |         |               |       |         |

- Step 7** Specify the following permissions for the selected user or group.
- **Read:** The user or group can read the object or list the contents of a directory
  - **Write:** The user or group can write to the object
  - **Execute:** The user or group can execute the object
  - **Delete:** The user or group can delete the object

In the Permissions table, the Current column displays the current value for the user or group's permissions. To change a permission, select a new value for it from the New column:

- Allow: The user or group can perform the specified action on the object.
- Deny: The user or group cannot perform the specified action on the object.
- Unset: The user's or group's permissions for this action on this object have not been specified; note that if you set all of an object's permissions to Unset and the user is not the ACL's owner, the user will be removed from the ACL.
- As is: The permission's value remains unchanged.

To submit the new permission values, click the **Submit** button.

- Step 8** Tell the consumer domain's administrator the data catalog path of the object to which access has been granted.



## Exposing users in a two-way interconnection

If you have a two-way connection with a remote Avaki domain, you might want to enable a trusted administrator of the remote domain to view users in your domain and add them to groups in the remote domain. This is an alternative to using interconnection IDs.

To do this, you must give the administrator permission to view and edit those user accounts, as follows:

**Step 1** Obtain the interconnect ID that represents the administrator of the remote domain.

**Step 2** Navigate to the Select User screen:

Home > User management > View and modify users

**Select User**

To modify a user's personal information, download a user interconnect ID, or change a password, select the user and click the **View/Edit** button. To delete a user from a grid domain, select the user and click **Delete All Checked**. To configure the user ID (UID) that is mapped to a user account, select the user and click **Update UIDs** below.

If you have a two-way connection with a remote domain, you may want to enable the administrator of the remote domain to view users in your domain and add them to groups in the remote domain. To do this, you must give the administrator permission to view and edit those users, as follows:

1. Obtain the interconnect ID that represents the administrator of the remote domain.
2. In the table below, select the users whose settings you want to allow the remote administrator to modify.
3. Go to the [Expose Users](#) screen to continue.

**Note:** The remote administrator will be able to edit only the users you selected, but the administrator will be able to view *all* the users in your domain.

**All users**

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

| Select                   | Username      | Authentication Service | View/Edit User            | Attributes                 | Security                 |
|--------------------------|---------------|------------------------|---------------------------|----------------------------|--------------------------|
| <input type="checkbox"/> | Administrator | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | MessagingUser | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | cmagadieu     | DefaultAuthService     | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

**Total number of users in domain: 3**

**Step 3** Select the users that you want to allow the remote administrator to modify.

**Step 4** Click the Expose Users link. The Expose Users screen appears

### Expose Users

You have chosen to let an administrator in a remote domain view all the users in your domain and modify the settings for the following users:

| Username | Authentication Service | Authentication Service Type |
|----------|------------------------|-----------------------------|
| wilma    | DefaultAuthService     | Grid                        |

To continue, click **Browse** and navigate to the directory where you stored the remote administrator's interconnect ID. Select the interconnect ID, then click **Submit**.

File name:

**Step 5** Click **Browse** and navigate to the directory where you stored the remote administrator's interconnect ID. Select the interconnect ID, then click **Submit**.

Now, when the administrator of the other domain accesses the Add User to Group page, the administrator will be able to add the users you've exposed to groups in the other domain.

## Viewing interconnected domains

To view a description of an Avaki domain to which your domain is interconnected, do the following:

- Step 6** Navigate to the View and Modify Interconnection screen:

Home > Domain interconnection management > View and modify interconnected grid domains

**View and Modify Interconnection**

You are connected to the following grid domain(s). Click **View Description** to view a description of a domain or click **Disconnect** to disconnect from a domain.

| Grid Domain Name | View Description                 | Configure Messaging                 | Disconnect                 |
|------------------|----------------------------------|-------------------------------------|----------------------------|
| Stephville       | <a href="#">View Description</a> | <a href="#">Configure Messaging</a> | <a href="#">Disconnect</a> |

- Step 7** Click the **View Description** link beside the name of the Avaki domain you want to view details about. The View Interconnection Description screen appears.

**View Interconnection Description**

The description for the grid domain **Stephville** is as follows:

**Description:**  
Stephville domain

## Enabling cross-domain messaging

If you add a remote domain's MessagingUser to your domain's MessagingUsers group, you can receive *update notifications* from that domain. This is called cross-domain messaging. When cross-domain messaging is enabled, generated views in your Avaki domain can be updated automatically when a particular event (usually the update of another generated view) occurs in the remote domain.

To enable cross-domain messaging, do the following:

- Step 1** Make sure that the remote domain has established an interconnect to your domain and has sent you the interconnect ID for the remote domain's MessagingUser, an internal system user.

- Step 2** Navigate to the View and Modify Interconnection screen:

Home > Domain interconnection management > View and modify interconnected grid domains

**View and Modify Interconnection**

You are connected to the following grid domain(s). Click **View Description** to view a description of a domain or click **Disconnect** to disconnect from a domain.

| Grid Domain Name | View Description                 | Configure Messaging                 | Disconnect                 |
|------------------|----------------------------------|-------------------------------------|----------------------------|
| Stephville       | <a href="#">View Description</a> | <a href="#">Configure Messaging</a> | <a href="#">Disconnect</a> |

- Step 3** Click the **Configure Messaging** link beside the name of the Avaki domain for which you want to configure messaging. The Configure Cross-Domain Messaging screen appears.

**Configure Cross-Domain Messaging**

The remote domain **Stephville** is currently not configured to send messages to your local domain. To allow **Stephville** to send messages to your domain, make sure that **Stephville** has established an interconnect to your domain. Enter the path to an interconnect ID for the messaging user in **Stephville** below, then click **Submit**.

File name:

- Step 4** Enter the path to the remote domain's MessagingUser user interconnection ID.

- Step 5** Click **Submit**. The MessagingUser is added to your domain's MessagingUsers group.

See the *Avaki Provisioning and Advanced Data Integration Guide* for information about configuring generated views to be updated automatically when a particular event occurs.

## Disabling cross-domain messaging

To disable cross-domain messaging and prevent a remote Avaki domain from sending messages to your domain, do the following:

**Step 1** Navigate to the Select Group page.

Home > User management > View and modify groups

**Select Group**

All groups

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

To modify a group's description, download a group interconnect ID, or add or remove group members, select the group and click the **View/Edit** button. To delete a group from a grid domain, select the group and click **Delete All Checked**. To configure the group ID (GID) that is mapped to a group account, select the group and click **Update GIDs** below.

| Select                   | Group name             | Authentication service | Authentication service type | View/Edit Group           | Attributes                 | Security                 |
|--------------------------|------------------------|------------------------|-----------------------------|---------------------------|----------------------------|--------------------------|
| <input type="checkbox"/> | Administrators         | DefaultAuthService     | Grid                        | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | DataProviders          | DefaultAuthService     | Grid                        | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | DatabaseAdministrators | DefaultAuthService     | Grid                        | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | DomainUsers            | DefaultAuthService     | Grid                        | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | MessagingUsers         | DefaultAuthService     | Grid                        | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | UserAdministrators     | DefaultAuthService     | Grid                        | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |
| <input type="checkbox"/> | WaterBuffaloLodge      | DefaultAuthService     | Grid                        | <a href="#">View/Edit</a> | <a href="#">Attributes</a> | <a href="#">Security</a> |

- Step 2** Click the **View/Edit** link beside the MessagingUsers group name. The Modify Groups screen appears.

**View and Remove Group Members**

All users in group

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

In the Select column, click the box beside the username and click **Submit** to remove a user from the group.

| Select                   | Username      | Domain       | Authentication Service |
|--------------------------|---------------|--------------|------------------------|
| <input type="checkbox"/> | MessagingUser | CherylDomain | DefaultAuthService     |
| <input type="checkbox"/> | MessagingUser | Stephville   | DefaultAuthService     |
| <input type="checkbox"/> | nagios        | Stephville   | myNIS                  |
| <input type="checkbox"/> | sjackson      | Stephville   | myNIS                  |

- Step 3** Place a check mark beside the MessagingUser username for the remote domain.
- Step 4** Click **Submit**. The remote domain's MessagingUser is removed from your domain's MessagingUsers group.

## Disconnecting interconnected domains

To disconnect your Avaki domain from another domain, do the following:

- Step 1** Navigate to the View and Modify Interconnection screen:

Home > Domain interconnection management > View and modify interconnected grid domains

**View and Modify Interconnection**

You are connected to the following grid domain(s). Click **View Description** to view a description of a domain or click **Disconnect** to disconnect from a domain.

| Grid Domain Name | View Description                 | Configure Messaging                 | Disconnect                 |
|------------------|----------------------------------|-------------------------------------|----------------------------|
| Stephville       | <a href="#">View Description</a> | <a href="#">Configure Messaging</a> | <a href="#">Disconnect</a> |

- Step 2** Click the **Disconnect** link beside the name of the Avaki domain you want to disconnect. The system displays a confirmation screen.
- Step 3** Click **OK** to confirm the operation. The remote domain is disconnected from your domain.





# Setting up log properties files

---

This appendix provides information to help you set up properties files for logging:

- “[Server log properties files](#),” below, explains how to configure the server logs maintained by each Avaki server.
- “[Request log properties files](#)” on [page 328](#) explains how to configure the request logs maintained by proxy servers and grid servers.

---

## Server log properties files

Avaki uses log4j, a logging package that lets you enable, disable, and configure logging with XML log properties files. A log4j properties file controls logging for each Avaki server and each instance of Avaki Studio. You can edit the log properties files to configure audit logging, logging for monitor tests and data services, and other features.

**Location of log properties files.** There is a log properties file for each Avaki server, and one for Avaki Studio. The properties files are located as follows, where `<Avaki_install_dir>` is the local directory into which Avaki software was installed:

```
<Avaki_install_dir>/jboss/server/grid-server/conf/log4j.xml
<Avaki_install_dir>/jboss/server/proxy-server/conf/log4j.xml
<Avaki_install_dir>/jboss/server/share-server/conf/log4j.xml
<Avaki_install_dir>/dgas_log.xml
<Avaki_install_dir>/avaki_studio/log4j.xml
```

---

**Note** See “[Logging](#)” on page 107 for the locations of server logs, request logs, and audit logs.

**For information about using log4j**, consult log4j documentation. This section describes log4j appenders and categories provided by Sybase, but it does not provide basic instructions on how log4j works or how to write or modify an appender. Here are some places to look for log4j documentation:

- Some online documentation is available on the Web:
  - API documentation: [logging.apache.org/log4j/docs/api/index.html](http://logging.apache.org/log4j/docs/api/index.html)
  - A short manual: [logging.apache.org/log4j/docs/manual.html](http://logging.apache.org/log4j/docs/manual.html)
- You can purchase a complete manual at [jakarta.apache.org/log4j](http://jakarta.apache.org/log4j).

**In this section:**

- “[Avaki logging categories](#),” below
- “[Configuring audit logging](#)” on page 320
- “[Sample log properties file](#)” on page 323

## Avaki logging categories

Avaki defines a number of loggable events and assigns each event to a category to make the events easier to manage. By default, all Avaki categories are enabled and except where noted, all events are recorded in server logs. You can configure log4j to send log entries for particular categories elsewhere (to a different log file or a database, for example, or as e-mail notifications). Or you can disable logging of particular categories altogether.

---

**Note** Logging categories are unrelated to the data catalog’s categories feature.

This table lists and describes the Avaki logging categories.

| Logging category                                | Description                                                                                                                                                                              |
|-------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| audit                                           | Controls logging of auditable events. For more information, see <a href="#">“Configuring audit logging” on page 320</a> .<br><br><b>Note:</b> The audit category is disabled by default. |
| Backup                                          | Controls logging of events related to backup.                                                                                                                                            |
| cacheable<br>cacheservice<br>com.avaki.dg.cache | Control logging of events related to caching and cache services.                                                                                                                         |
| Categories                                      | Controls logging of events related to categories in the data catalog.                                                                                                                    |
| com.avaki.dg                                    | Controls logging of events related to data grid access servers (DGASes).                                                                                                                 |
| Comm                                            | Controls logging of events related to network and grid communications and disk I/O.                                                                                                      |
| Database                                        | Controls logging of events related to the internal databases of Avaki servers and persistence of internal metadata.                                                                      |
| fr                                              | Controls logging of events related to failure resilience (failover). See also Replication.                                                                                               |
| Interactive                                     | Controls logging of interactive output events such as creating a view generator.                                                                                                         |
| io.view                                         | Controls logging of events related to Avaki data services and view generators.                                                                                                           |
| Javascript                                      | Controls logging of events related to JavaScript.                                                                                                                                        |
| jdbc                                            | Controls logging of events related to JDBC.                                                                                                                                              |
| Messaging                                       | Controls logging of events related to internal messaging.                                                                                                                                |
| Monitoring                                      | Controls logging of events related to active monitoring.                                                                                                                                 |
| Naming                                          | Controls logging of events related to naming and binding Avaki objects (in the data catalog and elsewhere in the system).                                                                |
| Replication                                     | Controls logging of events related to replication of servers (failover). See also fr.                                                                                                    |
| Search                                          | Controls logging of events related to search.                                                                                                                                            |

| Logging category | Description                                                                                                                                                                                                                                           |
|------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Security         | Controls logging of events related to security, including access control and Kerberos.                                                                                                                                                                |
| SQLView          | Controls logging of events related to SQL views.                                                                                                                                                                                                      |
| Studio           | Controls logging of events related to Avaki Studio.                                                                                                                                                                                                   |
| Syntax           | Controls logging of events related to command line syntax problems.                                                                                                                                                                                   |
| System           | Controls logging of system events such as initialization errors.                                                                                                                                                                                      |
| tds              | Controls logging of events related to TDS, an internal communications protocol.                                                                                                                                                                       |
| Transaction      | Controls logging of events related to distributed transactions and XA.                                                                                                                                                                                |
| TypeMapping      | Controls logging of events related to the mapping of data types during the creation of provisioned or generated tables. For information on mapping of data types, see the appendix on data type mapping files in the <i>Avaki Command Reference</i> . |
| VirtualDatabase  | Controls logging of events related to the Avaki virtual database.                                                                                                                                                                                     |
| VSM              | Controls logging of events related to metadata modeling (a.k.a. virtual schema modeling).                                                                                                                                                             |

## Configuring audit logging

Avaki's audit logging feature records grid events (such as the creation or execution of a data service) in a common store (such as a file or a database). Audit logging is disabled by default; you can enable it on individual grid servers, share servers, and data grid access servers (DGASes). Because nearly all auditable events occur on grid servers, however, you can get most of the benefits by enabling audit logging on your domain's grid servers. You can enable audit logging on proxy servers, but there is no point in doing so as no auditable events occur on proxy servers.

If you want to enable audit logging, you have several options: you can capture grid events in the audit log; capture grid events in the audit log, in the server log, and at the console; or write your own appender to send events to the common store(s) of your choice. Each of these options is discussed below. Loggable grid events are listed in [“Events captured by audit logging” on page 322](#).

**Capture grid events only in the audit log.** Avaki provides audit appenders in the log4j properties file for each grid server, share server and DGAS, but they are commented out. To enable the audit log, follow these steps:

**Step 1** Find the following lines in the server's log4j.xml or dgas\_log.xml file:

```
<!--
<appender name="AUDITFILE"
class="org.jboss.logging.appender.DailyRollingFileAppender">
 <param name="File" value="\${jboss.server.home.dir}/log/audit.log"/>
 <param name="Append" value="true"/>
 <param name="DatePattern" value="'. 'yyyy-MM-dd"/>
 <layout class="com.avaki.core.command.ConsoleLayout">
 <param name="PrintStackTrace" value="false"/>
 <param name="ConversionPattern" value="%d{ISO8601} %-5p [%c, %t] %m%n"/>
 </layout>
</appender>
-->

<!--
<category name="audit" additivity="false">
 <priority value="INFO"/>
 <appender-ref ref="AUDITFILE"/>
</category>
-->
```

**Step 2** Remove the comment lines before (<!--) and after (-->) each of the two sections.

**Step 3** Find the following lines in the server's log4j.xml or dgas\_log.xml file:

```
<!-- Suppress auditing categories. -->
<category name="audit" >
 <priority value="OFF"/>
</category>
```

**Step 4** Comment the lines out, like so:

```
<!-- Suppress auditing categories. -->
<!--
<category name="audit" >
 <priority value="OFF"/>
</category>
-->
```

**Capture grid events in the audit log, in the server log, and at the server console.** Do the following:

**Step 1** Follow the steps under “[Capture grid events only in the audit log](#),” above.

**Step 2** Find the following line in the server’s log4j.xml or dgas\_log.xml file:

```
<category name="audit" additivity="false">
```

**Step 3** Change the value of the additivity attribute to true.

**Write your own auditing appender** using log4j. This is a good choice if you want to record grid events (or a subset of grid events) in alternate common stores. All auditing appenders must use the audit categories defined in the Events.properties file, which is located in the Avaki installation directory:

```
<Avaki_install_dir>/docs/Events.properties
```

## Events captured by audit logging

The grid events listed here can be captured by audit logging.

Events Pertaining to Changes in ACLs	DGAS Event
audit.chmod	audit.accessDeniedByDGAS
audit.chown	
audit.chgrp	
Cache Association Events	Events Pertaining to HTTP Operations
audit.NodeCacheAssociate	audit.httpOpCreate
audit.DgasCacheAssociate	audit.httpOpExecute
	audit.httpOpDelete

<b>Data Catalog Events</b>	<b>Scheduling Events</b>
audit.fileCreate	audit.jobStatusUpdate
audit.fileDelete	audit.scheduler.delete
audit.folderCreate	audit.scheduler.update
audit.folderDelete	audit.scheduler.add
audit.fileReadAccess	
audit.fileReaddirAccess	
audit.fileWriteAccess	
audit.accessDeniedByDgas	audit.unbind
audit.bindPrimary	
audit.bindSecondary	
audit.catalogLookup	
audit.cifsAuthentication	
<b>Events Pertaining to Database Operations</b>	<b>User Events</b>
audit.dbopCreate	audit.userCreate
audit.dbopDelete	audit.userDelete
audit.dbopExecute	audit.userAddedToGroup
audit.dbopFWProxyExecute	audit.userRemovedFromGroup
audit.dbopCacheHit	audit.userAuthenticated
	audit.userAuthenticationFailed
<b>Events Pertaining to Data Services</b>	<b>Events Pertaining to Avaki View Generators and Generated Views</b>
audit.DSExecute	audit.viewCreate
audit.DSFWProxyExecute	audit.viewDelete
audit.DSCacheHit	audit.viewExecute
audit.DSCreate	audit.viewDependencyEvent
audit.DSDelete	
audit.DSDependencyEvent	

## Sample log properties file

This is a sample of the log4j properties file that controls monitor test logging on a grid server or a grid domain controller. Lines preceded by <!-- are comments.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">

<!-- ===== -->
<!-- -->
<!-- Log4j Configuration -->
<!-- -->
```

```

<!-- ===== -->
<!-- $Id: log4j.xml,v 1.5.2.6 2002/09/27 22:29:24 patriot1burke Exp $ -->
<!--
 | For more configuration information and examples see the Jakarta Log4j
 | owebsite: http://jakarta.apache.org/log4j
-->

<log4j:configuration xmlns:log4j="http://jakarta.apache.org/log4j/"
debug="false">

 <!-- ===== -->
 <!-- Preserve messages in a local file -->
 <!-- ===== -->

 <!-- A time/date based rolling appender -->
 <appender name="FILE"
class="org.jboss.logging.appender.DailyRollingFileAppender">
 <param name="File" value="{jboss.server.home.dir}/log/server.log"/>
 <param name="Append" value="true"/>

 <!-- Rollover at midnight each day -->
 <param name="DatePattern" value="'. 'yyyy-MM-dd"/>

 <layout class="com.avaki.core.command.ConsoleLayout">
 <param name="PrintStackTrace" value="false"/>
 <param name="ConversionPattern" value="%d{ISO8601} %-5p [%c, %t] %m%n"/>
 </layout>
</appender>

 <!-- ===== -->
 <!-- Preserve audit messages in a local file -->
 <!-- ===== -->

 <!-- To send audit events to the audit.log file, but exclude them
 from the console and the server.log file, uncomment the two
 sections directly below. Then comment out the "Suppress auditing categories"
 section later in this file.

 If you wish to send audit events to all appenders (audit.log, server.log,
 and the console), follow the steps above and then set the additivity attribute
 on the audit category to true.
-->

 <!--
 <appender name="AUDITFILE"
class="org.jboss.logging.appender.DailyRollingFileAppender">

```



```

<param name="File" value="\${jboss.server.home.dir}/log/audit.log"/>
<param name="Append" value="true"/>
<param name="DatePattern" value=".'yyyy-MM-dd"/>
<layout class="com.avaki.core.command.ConsoleLayout">
 <param name="PrintStackTrace" value="false"/>
 <param name="ConversionPattern" value="%d{ISO8601} %-5p [%c, %t] %m%n"/>
</layout>
</appender>
-->

<!--
<category name="audit" additivity="false">
 <priority value="INFO"/>
 <appender-ref ref="AUDITFILE"/>
</category>
-->

<!-- ===== -->
<!-- Append messages to the console -->
<!-- ===== -->

<appender name="CONSOLE" class="org.apache.log4j.ConsoleAppender">
 <param name="Target" value="System.out"/>

 <layout class="com.avaki.core.command.ConsoleLayout">
 <!-- The default pattern: Date Priority [Category] Message\n -->
 <param name="ConversionPattern" value="%d{ISO8601} %-5p [%c, %t] %m%n"/>
 <param name="PrintStackTrace" value="false"/>
 </layout>
</appender>

<!-- ===== -->
<!-- Put startup messages in a special file -->
<!-- ===== -->
<appender name="STARTUP-FILE" class="org.apache.log4j.FileAppender">
 <param name="File" value="\${jboss.server.home.dir}/log/connectinfo.txt"/>
 <param name="Append" value="false"/>

 <layout class="org.apache.log4j.PatternLayout">
 <param name="ConversionPattern" value="%d{ISO8601} %m%n"/>
 </layout>
</appender>

<category name="System.StartupInfo">
 <priority value="INFO"/>
 <appender-ref ref="STARTUP-FILE"/>
</category>

```

```

<!-- ===== -->
<!-- Sample e-mail notification configuration -->
<!-- ===== -->
<!--
<appender name="EMAIL" class="org.apache.log4j.net.SMTPAppender">
 <param name="BufferSize" value="512"/>
 <param name="SMTPHost" value="SMTPServer.domain.com"/>
 <param name="From" value="from@domain.com"/>
 <param name="To" value="to@domain.com"/>
 <param name="Subject" value="Avaki System Notification"/>
 <layout class="org.apache.log4j.PatternLayout">
 <param name="ConversionPattern"
 value="Timestamp: %d{ISO8601}%n%m%n%n"/>
 </layout>
 <filter class="org.apache.log4j.varia.LevelRangeFilter">
 <param name="LevelMin" value="ERROR"/>
 <param name="LevelMax" value="FATAL"/>
 </filter>
</appender>
-->

<!-- Enable this category to recieve email notifications when -->
<!-- messaging and scheduled event failures occur. -->
<!--
<category name="Messaging">
 <priority value="ERROR"/>
 <appender-ref ref="EMAIL"/>
</category>
-->

<!-- ===== -->
<!-- Limit categories -->
<!-- ===== -->

<!-- Limit Naming messges to ERROR -->
<category name="Naming">
 <priority value="ERROR"/>
</category>

<!-- Limit Monitoring messges to ERROR -->
<category name="Monitoring">
 <priority value="ERROR"/>
</category>

<category name="cacheservice">
 <priority value="INFO"/>
</category>

```

```

<category name="cacheable">
 <priority value="INFO"/>
</category>

<!-- suppress transaction manager warning that message service causes -->
<category name="org.jboss.resource.connectionmanager.TxConnectionManager">
 <priority value="ERROR"/>
</category>

<!-- Suppress auditing categories. -->
<category name="audit" >
 <priority value="OFF"/>
</category>

<!-- limit EJB deployment warnings to ERROR -->
<category name="org.jboss.ejb.EJBDeployer">
 <priority value="ERROR"/>
</category>
<category name="org.jboss.ejb.plugins">
 <priority value="WARN"/>
</category>

<!-- limit Jetty spew to warnings -->
<category name="org.jboss.jetty">
 <priority value="WARN"/>
</category>

<!-- ===== -->
<!-- Setup the Root category -->
<!-- ===== -->

<root>
 <priority value="INFO"/>
 <appender-ref ref="CONSOLE"/>
 <appender-ref ref="FILE"/>
</root>

</log4j:configuration>

```

## Request log properties files

Request log properties files are XML files that control HTTP request logging for proxy server and grid servers. Grid server request logs are for UI information only; proxy server request logs cover traffic through the proxy server.

The request log properties files are located in the following places (<Avaki\_install\_dir> is the local directory into which Avaki software was installed):

```
<Avaki_install_dir>/jboss/server/proxy-server/deploy/jboss-
web.sar/META-INF/jboss-service.xml
```

```
<Avaki_install_dir>/jboss/server/grid-server/deploy/jboss-
web.sar/META-INF/jboss-service.xml
```

To configure the request log for a proxy server or grid server, edit the appropriate jboss-service.xml file. In the file, look for the section headed “Configure the Request Log” (see below).

---

**Note** We recommend that you avoid editing any other section of this file, as the results may not be what you expect.

```

 <!--
===== -->
 <!-- Configure the Request Log
-->
 <!--
===== -->
 <Set name="RequestLog">
 <New class="org.mortbay.http.NCSARequestLog">
 <Arg><SystemProperty
name="jboss.server.home.dir"/><SystemProperty
name="jetty.log" default="/log"/>/yyyy_mm_dd.request.log
 </Arg>
 <Set name="retainDays">90</Set>
 <Set name="append">true</Set>
 <Set name="extended">true</Set>
 <Set name="LogTimeZone">GMT</Set>
 </New>
 </Set>

```

These are the properties you can configure:

<b>Request log property</b>	<b>Description</b>
Log file name	The name of the request log file. The default name of the log file takes the form yyyy_mm_dd.request.log, where yyyy_mm_dd is the date for which the log file was created. If the yyyy_mm_dd string is included in the file name, the log file rolls over every night and a new log file is created.
retainDays	The number of days rolled-over log files will be saved. Default: 90
append	If the server finds an existing log file when it starts, append new entries to the existing file rather than starting a new one. Default: true
extended	Use NCSA extended format for log entries. (Extended format logs include fields for referrals, user agents, and cookies.) Default: true
LogTimeZone	The time zone to use for the timestamps on log entries. Default: GMT (Greenwich mean time)



# *DGAS access effects*

---

This appendix describes the manner in which an Avaki data grid access server (DGAS) accesses data when it receives requests from an NFS client. A DGAS enables users to access files in a data grid from a machine running an ordinary NFS client. The DGAS imports and maintains user mappings between NFS user IDs (UIDs) and Avaki grid users. These mappings determine the Avaki user credentials that an NFS client user obtains during NFS-based access. The DGAS executes the user's request with the mapped credentials obtained. An NFS client may request the DGAS to change permissions and ownership of files as well, in which case the DGAS translates the NFS requests to corresponding grid requests.

Several of the actions the DGAS performs may not have immediately-obvious effects even though they satisfy the user's intent. In this appendix, we discuss the less obvious effects of NFS requests to the DGAS. Although we discuss NFS-based access predominantly, CIFS-based access is similar in several respects, but considerably simpler. This appendix covers the following topics:

- [“Host admission policy” on page 332](#)
- [“Mapping overrides in a DGAS” on page 333](#)
- [“Default user, group, UID and GID” on page 333](#)
- [“Changing permissions and ownership” on page 334](#)
- [“DGAS mapping cache” on page 336](#)
- [“Client attribute caching” on page 336](#)

- [“Access using older NFS clients” on page 337](#)
- [“CIFS-based access” on page 338](#)

---

## **Host admission policy**

A DGAS accepts NFS connect requests from only those machines listed in its host admission policy. Each entry in the host admission policy indicates whether a machine or a subnet is permitted or denied connection to the DGAS. If a particular machine or subnet is permitted to connect to the DGAS, an additional specification is the name of an authentication service in the grid to be used for user mapping. Together, these specifications indicate to the DGAS the machines from which NFS user requests will arrive, and the authentication service to use for mapping the NFS UID in those requests to grid user names.

Every authentication service in an Avaki domain maintains or can be configured to maintain an NFS UID for every user. Every authentication service also permits every DGAS in the same grid domain to query the service for mappings and user credentials. In rare cases, such as in interconnected domains, a DGAS may not have permissions to query an authentication service by default. In such cases, the administrator of the authentication service must grant read privileges on the authentication service to the administrator running the DGAS.

Although the Unix credentials within an NFS request consist of a UID, a primary GID and a list of GIDs for that UID, the DGAS ignores all of the GIDs. Instead, the DGAS uses the UID alone for impersonating the user to access files in an Avaki domain. GIDs are ignored because group membership is easy to change in Unix and can compromise grid-level security. Therefore, if a DGAS cannot determine whether to permit or deny an access based on the user, then it consults Avaki group membership, not Unix group membership, as the authority.



---

## Mapping overrides in a DGAS

A DGAS can be configured to maintain per-machine mappings that override mappings in an authentication service. For example, suppose a DGAS permits NFS connections from the machine 192.16.10.34, and associates the authentication service myNis with it. When users logging in to 192.16.10.34 issue NFS requests to the DGAS, the DGAS will map the UIDs associated with these requests as per the mappings in myNis. Therefore, a user “Bob,” with UID 503 on this machine, will get mapped to the appropriate grid user in myNis. However, the DGAS may be configured to override the mapping for Bob. Thus, when Bob issues an NFS request, the DGAS could be made to map UID 503 to grid user “Alice.” Alice can be any grid user, not necessarily a user in myNis. This mapping can be performed only by a grid administrator, and has effect only when Bob accesses the grid from machine 192.16.10.34. Multiple such mappings may be maintained within any DGAS.

Mapping overrides are useful in scenarios where several authentication domains exist but the grid administrator does not wish to integrate each domain into the grid. Overrides are also useful in scenarios where an organization has several machines that are not part of any authentication domain. Users from such machines can be mapped safely to an integrated authentication service, thus including those machines within one authentication domain for grid access. Finally, mapping overrides are necessary to support Avaki Compute Grid 2.x. Compute grid users may obtain temporary UIDs on compute machines for the duration of their jobs. DGAS mapping overrides enable such users to access their input and output files securely.

---

## Default user, group, UID and GID

An authentication service can be configured with a default user name so that NFS client users who are not mapped in a DGAS can continue to access the grid. When integrating an authentication domain into a data grid, the grid administrator may choose to import only a subset of the users in that domain. Consequently, users not imported into the grid cannot be mapped by any DGAS associated with this authentication service. Such users are denied access to the grid. However, the grid administrator may choose to let such unmapped users access the grid with the restricted privileges associated with one of the imported users. In such case, the administrator can configure the authentication service with a default user name. When a DGAS receives an NFS request from a user who has no mapping (i.e., no mapping is present in the authentication service *and* no mapping override is present in the DGAS), the DGAS checks the authentication service to see if a default user name is present. If it is, the DGAS maps the UID of the NFS request to this default user name and caches this mapping.

In Unix, only the root user is permitted to issue the **mount** command to connect to an NFS server such as a DGAS. Typically, root is not imported in an authentication service. However, in order to issue the mount, root must have minimal read privileges in the data grid. Therefore, setting a default user name for the authentication service enables root to obtain sufficient privileges to issue the mount.

Setting a default group name instead of a default user name has a similar effect. When the DGAS receives a request with an NFS GID that has no mapping, the DGAS checks the authentication service to see if a default group name is present. If it is, the DGAS maps the GID of the NFS request to this default group name and caches this mapping. This cached mapping is used only when determining the primary group of the owner during permission and ownership changes.

An authentication service can also be configured with a default UID and GID so that the DGAS can display proper ownership and permissions of files owned by unmapped users. In contrast to the default user/group names, which are used to permit/deny access, the default UID/GID is used for display alone, and on most operating systems does not affect access. When an NFS client issues a request that requires displaying file ownership and permissions, the DGAS consults its user mappings to determine what to display. However, in some cases, a file being displayed may be owned by a grid user who is not mapped in the authentication service associated with the DGAS. In that case, the DGAS displays the file as being owned by user root and group root. Instead, if the authentication service is configured with a default UID and GID, the same file will be displayed as being owned by that UID and GID. When the default UID/GID are used, the DGAS continues to display the true permissions of the file.

The default UID/GID does not affect access from the DGAS. However, some NFS clients (notably, on SGI) may deny permission based on the displayed UID/GID. In general, we recommend ensuring that all Avaki users have a UID mapping, if possible.

---

## ***Changing permissions and ownership***

Grid-level file permissions and ownership can be changed by a user accessing a DGAS from an NFS client. The process of changing permissions and ownership is consistent with Unix even though Avaki ACLs are richer than Unix permissions. A Unix file permission has three possible actions: read, write and execute. In contrast, a file in an Avaki grid has four possible actions: read, write, execute and delete. For each action, Unix provides two choices: set and unset. Avaki provides three choices: allow, deny and unset. When permissions are displayed and changed from an NFS client, the DGAS ensures that reasonable and equivalent grid permissions are changed.

As in Unix, changing permissions can be accomplished only by the owner of the file—in this case, the grid user to whom the UID in the NFS request maps. When a permission change is requested, the DGAS changes a series of ACLs on the grid file based on the new mode sent. These ACL changes may affect the file permissions for the owner, the primary group and the grid user “Everyone.” Expectedly, changing read and execute permissions from the NFS client changes the equivalent permissions in the grid, but changing write permissions from the NFS client changes write as well as delete permissions in the grid.

As in Unix again, changing ownership of a file can be performed only by the current owner of the file. If the DGAS property `unix-chown-semantic` is set to “True” (default), the DGAS will revoke the old owner’s permissions and transfer them to the new owner. If this property is “False,” the revocation and transfer will not occur unless the NFS client sends a subsequent change permissions command. Although a file in a grid may have group permissions, Avaki neither has nor needs the notion of a primary group as Unix does.

Under certain conditions, the NFS request to change permissions and ownership cannot be satisfied by the DGAS despite seemingly-correct privileges:

- The NFS request contains a UID that is not mapped in the authentication service, is not mapped within the DGAS itself, and cannot be mapped to any other user because the authentication service does not have a default user name configured.
- The NFS request contains an additional request to change groups (“chgrp” plus “chmod”), but the GID in the request is not mapped in the authentication service, is not mapped within the DGAS itself, and cannot be mapped to any other group because the authentication service does not have a default group name configured.
- The current owner of the file is not mapped in the authentication service, is not mapped within the DGAS itself, and cannot be mapped to any other user because the authentication service does not have a default user name configured.
- The NFS request contains an additional request to change groups (“chgrp” plus “chmod”), but the current primary group owner of the file is not mapped in the authentication service, is not mapped within the DGAS itself, and cannot be mapped to any other group because the authentication service does not have a default group name configured.

---

## DGAS mapping cache

During normal operation, a DGAS populates an in-memory LRU cache of user mappings as well as group mappings. This cache permits a DGAS to determine the mapping from an NFS UID to a grid user name quickly. If such a mapping is not present in memory either because it has been evicted or the NFS user has not been encountered previously, the DGAS contacts the authentication service to procure a mapping. This in-memory cache is purged and refreshed every two hours in order to capture changes in the authentication service. Alternatively, you can purge this cache from the web UI.

---

## Client attribute caching

Most NFS clients maintain a file mode/attribute cache that contains permissions and ownership information for recently-accessed files and directories as well as a content cache for files. These caches improve performance for frequently-accessed files. However, the contents of the file attribute cache can conflict with grid-level ACLs for short periods.

For example, suppose “Bob” currently has grid-level permissions to access the file “data.” When Bob accesses the file, the DGAS stores the file “data” in its own cache, the NFS client stores “data” in its content cache as well, and the NFS client stores Bob’s permissions in its attribute cache. Shortly thereafter, Bob loses grid-level permissions to “data.” However, he may be able to continue accessing “data.” The reason for this continued access is that the NFS client has cached Bob’s permissions and therefore allows him access to the file. The DGAS never receives the request to access the file, and therefore does not check permissions or retrieve the file from the authoritative source. The solution to this scenario is either to wait the short time it takes for the NFS client cache to purge itself, or start the NFS client with client attribute caching turned off (“actimeo=0”).

---

## Access using older NFS clients

When connecting to a DGAS from older NFS v2 client or even some NFS v3 clients (e.g., Microsoft SFU mount), the DGAS must be configured with Unix file mode semantics turned off in order to permit access. These NFS clients merely refer to the Unix file mode to permit or deny access. However, since Avaki access control lists (ACLs) are richer than Unix file permissions, it is often not possible to represent the correct grid-level permissions on a file to an NFS client. In order to compensate for this behavior on the part of NFS clients, the DGAS must be configured with the property `unix-file-mode-semantics` set to “False”. Doing so informs the DGAS to adjust the “other” part of the file permissions to permit access to users who are neither owners nor primary group members, but are part of the grid-level ACL. Setting this property to “False” is *not* a security risk—although the above NFS clients permit access to the file, the DGAS itself continues to mediate accesses based on the ACLs associated with the file.

Setting `unix-file-mode-semantics` to “False” and leaving Client Attribute Caching on can raise security concerns, such as permitting unauthorized access. Consider a case where “Bob” accesses a file called “data” not owned by him, but for which he has grid-level read permissions. Since Bob accesses “data” through an older NFS client, the grid Administrator sets `unix-file-mode-semantics` to “False” on the relevant DGAS. As a result, the DGAS correctly adjusts the “other” permissions on “data” to permit Bob access. However, the NFS client from which Bob accesses “data” has been started with Client Attribute Caching left on. Shortly after Bob’s access, “Alice,” who does not have grid-level permissions to read “data,” accesses it *from the same mount on the same NFS client* as Bob. Since the NFS client has cached the “other” permissions as well as the content for “data,” it will permit access to “data.” Note that the DGAS is not involved in this access at all because the NFS client does not forward Alice’s request. Consequently, the DGAS cannot prevent Alice’s unauthorized request.

For standard NFS v3 clients, setting `unix-file-mode-semantics` to the default “True” value ensures correct behavior from the DGAS as well as the NFS client. Since most NFS v3 clients do not rely on file mode semantics alone, the DGAS can mediate access based on grid-level ACLs correctly.

---

## **CIFS-based access**

User mapping for CIFS users is considerably simpler than for NFS users because CIFS does not have the concept of groups and file ownership. Moreover, it is not possible to change file permissions from a CIFS client. A CIFS client authenticates a request (also known as an SMB session) with the DGAS by passing a user name and using a challenge-response protocol. The DGAS then determines whether the authentication service associated with this machine in its host admission policy is of type Grid, NIS or LDAP. For Grid and NIS services, the DGAS requests the password of the user from the service, computes a challenge and issues the challenge to the client. For LDAP services, the DGAS passes the user name to a domain controller, requests it to compute a challenge and passes the challenge to the client. In either case, the client receives a challenge from the DGAS, computes a response and returns it to the DGAS. For Grid and NIS services, the DGAS checks the response for correctness. For LDAP services the DGAS passes the response to the Windows domain controller, which checks the response and informs the DGAS about its correctness. In either case, if the response is correct, the DGAS fetches the user's credentials from the authentication service and acts on behalf of the Avaki user corresponding to the CIFS client user.

# Upgrading to Avaki 7.0

---

This appendix explains how to upgrade an Avaki domain from Avaki version 6.2.1 to version 7.0.x. We recommend that the upgrade procedures be performed by a member of the Sybase Professional Services group.

If your Avaki domain is currently running a release earlier than 6.2.1, you must upgrade to the highest available EBF (patch version) of Avaki 6.2.1 before you upgrade to 7.0.x. For instructions on upgrading to 6.2, see the 6.2 version of the *Avaki Administration Guide*. For the latest EBF, go to <http://downloads.sybase.com>. Log in and select Avaki EII.

---

**Note** Be sure to review the readme file for this release. You can find it in <Avaki-install-dir/docs.

---

## Interoperability

Upgrade all the Avaki servers, command line clients, and Avaki Studio instances in an Avaki domain at the same time. Within a single Avaki domain, Avaki servers running version 7.0.x cannot interoperate with servers or clients (including command line clients and Avaki Studio) that are running earlier versions of Avaki software.

## Preparing to upgrade

If you're not sure what software version is running on your Avaki servers, use the **avaki upgrade --info** command to display that information. **avaki upgrade --info --all** displays version information for all grid servers and share servers in the Avaki domain. **avaki upgrade --info --server=<server-name>** displays version information for the specified grid server or share server.

In the procedures in this appendix,

`$AVAKI_OLD` refers to the directory on each machine where the older version of Avaki software is installed. (This might be `C:\AvakiDataGrid62`, for example.)

`$AVAKI_62_BACKUP` refers to the directory on each machine where you'll create a back-up copy of the Avaki install directory. (You might call this `C:\Avaki_62_Backup`—but you don't need to create it yet.)

`$AVAKI_NEW` refers to the directory where you'll install the newer version of Avaki software. (This might be `C:\AvakiDataGrid70`—but again, don't create it yet.)

These must be *separate* directories.

## Upgrade planning

Treat the upgrade of your Avaki domain as a formal project and understand that its success depends on the time, resources, and planning you devote to it. We strongly suggest that you prepare a written plan before starting the upgrade. Your upgrade plan should include:

- **Enough time to complete the upgrade process and test the upgraded domain.** The time required depends on the number, type, and configuration of the servers in your domain. Here are some rules of thumb on how much time to allow for each server:

Server type	Time to upgrade	Notes
Grid server including GDC	1 to 1.5 hours	Upgrade time varies based on the number of Avaki shares, database connectors, database operations, data services, and view generators on each grid server.



Server type	Time to upgrade	Notes
DGAS	30 minutes	
Share server, proxy server, and Avaki Studio	20 minutes	

- **An execution plan** that includes checklists of tasks for each Avaki server and general tasks. You can use the procedures in this appendix (“[Preparing your domain for the upgrade](#)” on page 341 and “[Upgrade procedures](#)” on page 343) as starting points for the checklists.
- **A test plan** that ensures that the upgraded domain is fit to return to service. See “[Testing](#)” on page 347.

Your test plan should include checks on a frequently run scheduled task on each grid server that hosts scheduled tasks. If your domain doesn’t have any scheduled tasks that run frequently, create a scheduled task that runs at 5-minute intervals so you can use it to verify that scheduled tasks are working after the upgrade.

## Preparing your domain for the upgrade

Follow these steps to prepare your Avaki domain for the upgrade:

- Step 1** Examine the log of each Avaki server in the domain. If you notice any problems (such as servers failing to communicate with one another), resolve them before continuing with the upgrade. For instructions on finding the log for a server, see “[Server logs](#)” on page 107.
- Step 2** On every machine that hosts an Avaki server, an Avaki Studio instance, or a command client, install the new Avaki software version, taking care to ensure that *the old and new installations are in different directories*. For installation instructions, see “[Installing in Unix](#)” on page 16 or “[Installing in Windows](#)” on page 22.

**Caution** Do not start any 7.0 servers until you are directed to do so in a later procedure. If you start servers too soon, the data in your installation directories may be compromised. To recover from a premature start, re-install the new software.

**Step 3** Copy any database drivers in the 6.2 <Avaki-install-dir>/drivers directory to the same location in the 7.0 installation:

```
$AVAKI_OLD/drivers/* to
$AVAKI_NEW/drivers/*
```

**Step 4** Terminate all grid activities by all users. For example, make sure all users are logged out from their web user interfaces, that all instances of Avaki Studio are disconnected from their servers, and that no user is accessing any file or directory on the grid from an NFS or CIFS DGAS client.

**Step 5** Unmount all the NFS clients of every DGAS in the Avaki domain.

**Step 6** Shut down all Avaki servers from the older installation in this order:

- DGASes and proxy servers (in any order)
- Share servers (these must all be shut down before you shut down any grid server)
- Ordinary grid servers
- Secondary GDC, if any
- Primary GDC

Refer to the 6.2 version of the *Avaki Administration Guide* for details on how to shut down each kind of server.

**Caution** Do not restart any 6.2 servers until you have completed all the upgrade procedures. If you restart servers too soon, the data in your installation directories may be compromised. To recover from a premature start, return to [Step 2](#) of this procedure and perform the upgrade again.

**Step 7** Unregister any Avaki servers that are registered as services (for auto-restart).

**Step 8** Make a back-up copy of the 6.2 Avaki installation directory on every machine in the Avaki domain. In the section that follows, this back-up directory is called \$AVAKI\_62\_BACKUP.

---

# Upgrade procedures

The procedures in this section explain how to upgrade your Avaki domain:

- [“Copying internal data,”](#) below
- [“Starting the upgraded servers”](#) on page 346

## Copying internal data

On each server in your Avaki domain, follow the steps in this procedure to copy internal databases, caches, bindings files and workspaces to the new installation and to configure system properties.

The purpose of copying the 7.0 and 6.2 db directories back and forth is to create a composite copy of the db directory that contains both of the following:

- your data and server configurations from the 6.2 installation, and
- the new metadata from the 7.0 installation.

**Step 1** Back up the db directories in the 7.0 installation directory:

```
$AVAKI_NEW/jboss/server/grid-server/db/* to
$AVAKI_62_BACKUP/jboss/server/grid-server/db/*

$AVAKI_NEW/jboss/server/proxy-server/db/* to
$AVAKI_62_BACKUP/jboss/server/proxy-server/db/*

$AVAKI_NEW/jboss/server/share-server/db/* to
$AVAKI_62_BACKUP/jboss/server/share-server/db/*
```

**Step 2** Copy the 6.2 db directories on top of the 7.0 db directories:

```
$AVAKI_OLD/jboss/server/grid-server/db/* to
$AVAKI_NEW/jboss/server/grid-server/db/*

$AVAKI_OLD/jboss/server/proxy-server/db/* to
$AVAKI_NEW/jboss/server/proxy-server/db/*

$AVAKI_OLD/jboss/server/share-server/db/* to
$AVAKI_NEW/jboss/server/share-server/db/*
```

**Step 3** Copy the saved 7.0 db directories back, overwriting any 6.2 files.

```
$AVAKI_62_BACKUP/jboss/server/grid-server/db/* to
$AVAKI_NEW/jboss/server/grid-server/db/*
```

```
$AVAKI_62_BACKUP/jboss/server/proxy-server/db/* to
$AVAKI_NEW/jboss/server/proxy-server/db/*
```

```
$AVAKI_62_BACKUP/jboss/server/share-server/db/* to
$AVAKI_NEW/jboss/server/share-server/db/*
```

**Step 4** Copy the contents of the caches directory from the 6.2 installation to the 7.0 installation:

```
$AVAKI_OLD/caches/* to
$AVAKI_NEW/caches/*
```

**Step 5** Copy the contents of the DGAS/dgas\_db directory from the 6.2 installation to the 7.0 installation:

```
$AVAKI_OLD/DGAS/dgas_db/* to
$AVAKI_NEW/DGAS/dgas_db/*
```

**Step 6** Copy any DGAS caches from the 6.2 installation to the 7.0 installation:

```
$AVAKI_OLD/DGAS/cache/* to
$AVAKI_NEW/DGAS/cache/*
```

**Step 7** Copy the contents of Avaki Studio's workspace directory from the 6.2 installation to the 7.0 installation. The following example assumes the default workspace location for both the old and the new installations:

```
$AVAKI_OLD/workspace/* to
$AVAKI_NEW/workspace/*
```

---

**Note** If you have configured a workspace directory outside the Avaki installation directory, you can simply enter its path when you start the new version of Avaki Studio—there's no need to copy the directory.

- Step 8** Copy the `bindings.xml` files for every grid server, including the GDC, from the 6.2 installation to the 7.0 installation. This is necessary because several of the default ports used by grid servers have changed in 7.0, but existing grids must continue to use the old default ports.

```
$AVAKI_OLD/jboss/server/grid-server/conf/bindings.xml to
$AVAKI_NEW/jboss/server/grid-server/conf/bindings.xml
```

- Step 9** If during the deployment of the older data grid any default ports were changed for the internal share servers on any grid servers, including the GDC, copy the following files for the affected servers from the 6.2 installation to the 7.0 installation:

```
$AVAKI_OLD/jboss/server/grid-server/conf/shareserver.ports to
$AVAKI_NEW/jboss/server/grid-server/conf/shareserver.ports
```

- Step 10** If during the deployment of the older data grid any default ports were changed on other Avaki servers, copy the following files for the affected servers from the 6.2 installation to the 7.0 installation:

```
$AVAKI_OLD/jboss/server/proxy-server/conf/bindings.xml to
$AVAKI_NEW/jboss/server/proxy-server/conf/bindings.xml
```

```
$AVAKI_OLD/jboss/server/share-server/conf/bindings.xml to
$AVAKI_NEW/jboss/server/share-server/conf/bindings.xml
```

```
$AVAKI_OLD/jboss/server/share-server/conf/shareserver.ports to
$AVAKI_NEW/jboss/server/share-server/conf/shareserver.ports
```

- Step 11** If during the deployment of the older data grid any system properties files or startup scripts were changed, make equivalent changes in the new grid. For example, if `$AVAKI_OLD/jboss/server/<server-type>/conf/system.properties` was changed in order to provide a specific DNS name for the machine, make an equivalent change to `$AVAKI_NEW/jboss/server/<server-type>/conf/system.properties`. See [“Setting a server’s host name or IP address” on page 131](#) for instructions.

---

**Note** The fully qualified hostname for each Avaki server in the upgraded data grid must be exactly the same as the hostname for the corresponding server in the old grid.

If `$AVAKI_OLD/share-server` was changed to increase the memory limit for the Java virtual machine, make an equivalent change to `$AVAKI_NEW/share-server`.

## Starting the upgraded servers

- Step 1** Start the grid servers from the 7.0 installation. Bring up the servers in the following order:
- The primary GDC
  - The secondary GDC, if any
  - All ordinary grid servers
- Step 2** On each grid server (including GDCs), check the server log to confirm that the upgrade worked—the log will include a message that says “Upgrade completed successfully.” If the success message is not in the log, contact Sybase technical support. (For instructions on finding server logs, see [“Server logs” on page 107.](#))
- Step 3** Shut down and restart all the grid servers (including GDCs). When you restart the servers, follow the same order used in [Step 1](#).
- Step 4** Start the share servers, DGASes and proxy servers from the 7.0 installation. Do not start any share servers until all the grid servers in the domain are running.

---

## Post-upgrade tasks

After you upgrade an Avaki domain, perform the following tasks.

1. Remount DGAS NFS clients. See [“NFS-mounting Avaki directories” on page 92.](#)
2. Reconnect command clients ([page 95](#)) and Avaki Studio (see *Data Integration with Sybase Avaki Studio*).
3. Test the upgraded domain (see [“Testing,”](#) below).
4. Resume grid activity. You can do this immediately, though you might choose to wait until the testing phase is complete.

---

# Testing

Follow your test plan to ensure that your upgraded Avaki domain is functioning properly. Your test plan should include the following:

- Look in the server logs for indications of the success or failure of the upgrade.
- Have multiple users (including several nonadministrative users and users from each authentication service) log in to the domain.
- Make sure that scheduled tasks (such as data service executions and share rehashes) are running, including any schedules you created to test the upgrade.
- Make sure you can create new schedules.
- Make sure you can access data in Avaki shares.
- Make sure remote and local caches are accessible and are being populated. (See the *Sybase Avaki EII Provisioning and Advanced Data Integration Guide* for instructions on viewing caches.)





---

# Glossary

---

Terms printed in *italics* are defined in the glossary.

## **access control list**

(ACL) A list, for a given file, directory, or other Avaki object, of permissions—read, write, execute, delete, and owner—that control which users and groups can view, modify, invoke, and remove the object, and edit the object's ACL.

## **ACL**

See *access control list*.

## **ad-hoc query**

A mechanism that lets you directly query a database in SQL. The query must run through an existing Avaki *database connector*. You can run an ad-hoc query using either the CLI or a *JDBC driver*. Ad-hoc queries can be thought of as single-use *database operations*.

## **attribute**

A property of an *Avaki directory*, file, *service*, or other object. Each attribute has a name, a type (string, integer, float, date, time, or timestamp) and a value. System attributes are read-only; you can change the values of other attributes. You can also create new attributes and add them to objects as needed.

## **authentication service**

A *service* associated with an *Avaki domain* that authenticates an Avaki user's identity and provides security credentials each time the user logs in. Avaki can be configured to use third-party directory services as authentication services for login; for user accounts created directly in the Avaki domain, Avaki uses its own default authentication service.

**Avaki directory**

Avaki software creates a single, unified namespace that is accessible (subject to Avaki *access control lists*) to all users in the *Avaki domain*. The namespace, called the *data catalog*, is arranged as a hierarchy of Avaki directories (folders). The catalog directory structure is stored by the domain's grid servers and its GDC, while the physical files remain in their original locations in your local file systems. When you work with directories, it's important to distinguish between Avaki directories, which are part of the data catalog, and local directories, which reside in your local file system.

**Avaki domain**

The basic administrative unit of the Avaki EII system. An Avaki domain consists, at a minimum, of one *grid domain controller* and may also include one or more *grid servers*, *share servers*, *proxy servers*, *data grid access servers*, and *command clients*. See also *domain name*.

**Avaki group**

A set of users who have the same permissions on one or more Avaki objects. You can use the group name in place of a user name when you set permissions or create *access control lists*.

**Avaki installation directory**

The directory in your local file system where Avaki software is installed. This is not a *data catalog* directory.

**Avaki share**

(Also shared directory.) A pointer in the Avaki *data catalog* to a directory or file in the underlying local file system. When you browse the data catalog, Avaki shares look like—and can be accessed like—other Avaki directories. Contrast with *CIFS share*.

**Avaki server**

A *service* that starts, stops, and monitors other Avaki services on a particular computer. Every server is part of an *Avaki domain*. A server is permanently attached to the computer where it is started. There are several types of server: *data grid access servers*, *grid domain controllers*, *grid servers*, *share servers*, and *proxy servers*.

**Avaki Studio**

A graphical, metadata-based data integration tool that lets you

- Build data flows by dragging and dropping input sources, operators, and output targets. You can deploy your data flows as *Avaki data services*.
- Import or create *metadata models* and apply them to Avaki objects or use them to build new data services.

In addition, you can use Studio to perform provisioning tasks (creating *database connectors*, *database operations*, *virtual database operations*, and *SQL views*), manipulate *categories*, and edit *ad-hoc queries* and *attributes*.

## cache service

(Formerly proxy cache service.) A staging service that stores copies of files, *database operation* results, and *data service* results. Caching improves retrieval performance. To ensure that an object is stored in the cache, you can *pin* a file or directory in the data catalog, or schedule a database operation or data service. A cache service can provide remote caching, local caching, or both. The freshness of cached data is controlled by a data expiration interval that determines how long cached data is considered valid and by a cache coherence window that tells the cache service how often to check whether cached data is still valid. If cached data is too old to satisfy a new request (or is not stored in this cache), the cache service does one of the following:

- If the database operation or data service that produced the data is local to this cache service, the cache service triggers execution of the database operation or data service.
- If the database operation or data service that produced the data is remote from this cache service, this cache service requests the data from the data source's local cache service.

A cache service can be associated with a *data grid access server*, a *grid server*, or a local user in a CLI session. See also *local cache*, *remote cache*, *on-demand caching*, and *scheduled caching*.

## category

A mechanism for classifying and organizing the contents of the *data catalog*. Like *Avaki directories*, categories serve as containers for objects in the data catalog. Anything in the data catalog—views, data services, shared files, even Avaki directories themselves—can be assigned to a category. Categories are hierarchical, they have attributes, and Avaki *access control lists* regulate access to them.

## CIFS client

A machine that mounts files or directories from the Avaki *data catalog* by connecting to a *CIFS share* through an Avaki *data grid access server*. A CIFS client need not have Avaki software installed. (CIFS—Common Internet File System—is a file-sharing protocol based on the file system implemented by Windows.)

## CIFS share

A directory or file that has been exported (shared) from the Avaki *data catalog*. A CIFS share can be mapped into a Windows file system like a network drive. When you browse the Windows file system, CIFS shares look like—and can be accessed like—other files and directories. CIFS shares are created through a *data grid access server*. Contrast with *Avaki share*.

**client**

Avaki supports several types of client: *Avaki Studio*, *CIFS clients*, *command clients*, *JDBC/ODBC clients*, *NFS clients*, *web clients*, and *WS clients*.

**command client**

A machine that can issue Avaki commands but does not contribute resources to the *Avaki domain*.

**connect port**

The connect port on a *grid domain controller*, *grid server*, *data grid access server*, *proxy server*, or *share server* accesses the JNDI naming service or RMI registry for the underlying application server. The connect port is one of many ports that a GDC or server uses to communicate with other Avaki objects. You must supply the connect port number of a target grid server or GDC whenever you connect a new object (another server, a copy of Avaki Studio, or a *command client*, for example) to an *Avaki domain*. When you *interconnect* two Avaki domains, you must supply each domain's connect port number to the other one.

**data catalog**

A hierarchical structure similar to a file system that encompasses all objects in an *Avaki domain*. The data catalog contains *Avaki directories* and files, *Avaki shares*, *Avaki servers*, *SQL views*, *database operations* and *data services*, and other objects.

**data grid access server**

(DGAS) An *Avaki server* that makes *Avaki directories* and their contents available to *CIFS clients* and *NFS clients*.

**data service**

An operation that transforms data obtained from sources in the *data catalog*. Input data can come from any number of sources, including:

- other data services
- data catalog files (which can be *generated views*)
- *Avaki database operations* (which in turn extract the data from relational databases)
- HTTP requests
- Web service invocations

You can generate the code that manipulates the data by creating a *view model* in *Avaki Studio*, or by writing a custom *data service plug-in* using Java, JavaScript, or XSLT. Data service output can be in rowset or XML format. Data services are run by the *execution services* on *grid servers*, they can be scheduled, and their results can be cached.

## data service plug-in

The logic for a *data service*, written in Java, JavaScript, or XSLT. Data service plug-ins are modular—you can use the same plug-in for multiple data services. *Avaki Studio* creates data services and plug-ins simultaneously, so if you use Avaki Studio to create data services, you don't have to worry about plug-ins. You can also use the Avaki Plug-in Wizard to create data service plug-ins.

## database connector

A mechanism that enables one or more *database operations*, *SQL views*, or *ad-hoc queries* to connect to a relational database.

## database operation

(DBOP) A mechanism that can

- extract data from a relational database and deliver it on demand to a *view generator* or a *data service*, or
- modify data in a relational database.

A database operation can be a SQL statement or a stored procedure call.

## dependency

A relationship in which an Avaki object requires input from other Avaki objects. A *data service* might require input from one or more *database operations* or from other data services. A *view generator* might depend on a database operation for input. A database operation can serve as an input source for one or more data services or view generators. Generated *SQL views* depend on database operations, virtual database operations, or data services. You can use *Avaki Studio*, the web UI, or the CLI to list input and output dependencies for any data service, database operation, or view.

## DGAS

See *data grid access server*.

## distributed transaction

A set of related operations (typically SQL operations such as SELECT, INSERT, UPDATE, DELETE, and CALL) that

- involve one or more databases, and
- might lead to unwanted results (such as leaving participating databases in an inconsistent state or producing inconsistent reads) if some of the operations complete and others do not, and therefore
- must all be executed at once, as a single transaction.

The individual operations that make up a distributed transaction are performed by *database operations* that use *database connectors* configured with XA-capable *JDBC drivers*; all the database operations are executed, using the two-phase commit protocol, by a specially configured *data service*. The two-phase commit protocol is designed to ensure that the participating databases will be left in a consistent state—that is, that all the operations in the distributed transaction will be completed, or none of them will.

**domain name**

A unique alphanumeric identifier for an *Avaki domain*. The domain name is assigned by the Avaki administrator when the Avaki domain is initialized. The domain name has a maximum length of 30 characters.

**enterprise information integration**

(EII) A software system that

- enables applications and users to access, without replication, both raw and integrated data from multiple heterogeneous distributed data sources while hiding the complexity of the data sources, and
- provides tools enabling users and data owners to further integrate and transform data.

**exclusion**

See *schedule exclusion*.

**execution service**

Execution services execute *data services*. There is an execution service on every *grid server*, and you can configure a pool of execution services for load-sharing. When a pool is in place, a data service can be run by any execution service in its grid server's pool.

**failover**

The transition of control from a failing or unreachable primary *grid domain controller* to a secondary grid domain controller.

**federated data access**

A scheme that allows independently controlled elements to be shared into a single namespace. Files, user accounts, and other objects maintain their separate identities and remain under the control of their owners, but—subject to access controls—the objects can be accessed, managed, and viewed as if they were part of a single system.

**GDC**

See *grid domain controller*.

**generated view**

A file created by a *view generator*; it may contain data obtained from a *database operation*, a *data service*, a file, or an HTTP source. Like other files, generated views exist in a local file system and are shared into the *data catalog*.

**grid**

A heterogeneous group of networked resources that appears and functions as one operating environment. A data grid like the Avaki Enterprise Information Integration (EII) system provides secure, shared access to data.

**grid directory**

See *Avaki directory*.

**grid domain**

See *Avaki domain*.

**grid domain controller**

(GDC) The first server in an *Avaki domain* is the grid domain controller. The GDC maintains a portion of the Avaki domain's namespace and provides authentication services. It can also run Avaki commands, share data, and monitor other servers. (That is, the GDC functions as a *grid server*.) If the domain is configured for *failover*, it has both a primary GDC and a secondary GDC; the secondary is updated at regular intervals and takes over management of the domain if the primary fails. Any Avaki shares managed by the primary are read-only on the secondary.

**grid server**

An *Avaki server* that maintains a portion of the *Avaki domain*'s namespace, runs Avaki services such as shares, execution services, caches, and searches, and allows you to run Avaki's web UI and execute Avaki commands.

**group**

See *Avaki group*.

**hard link**

Provides an alternate name for an item in the *data catalog*. Changes to the object's other names have no effect on the hard link: you can move or change a file's original name and the hard link will still know where to find the file. To delete a hard-linked object, you must remove its original name. Contrast with *soft link*.

**interconnect**

To create a unidirectional link from one *Avaki domain* to another. Interconnecting lets an Avaki domain make its *data catalog* visible to users in another domain (subject to Avaki access controls).

**JDBC driver**

JDBC (Java Database Connectivity) drivers allows application programmers to access database data shared in the *data catalog*. When a JDBC driver accesses data, it returns a JDBC result set that's immediately available to your program. JDBC drivers can:

- Call any *data service* in the data catalog
- Call any *database operation* in the data catalog
- Perform SQL `select` operations against *SQL views* in the data catalog

Sybase offers three JDBC drivers for use with Avaki EII software:

- The included Avaki JDBC driver
- jConnect, Sybase's standard JDBC driver
- An XA-capable driver for use with *database connectors* that support *distributed transactions*

**link**

See *hard link*, *soft link*.

**local cache**

A *cache service* that runs on the same *grid server* as a *database operation* or a *data service* that generates cachable data. The local cache stores results produced by local database operations and data services so they don't have to execute for every new request. See also *remote cache*.

**metadata model**

A construct in *Avaki Studio* that expresses a schema by defining a set of tables. A table in a metadata model can be mapped (linked) to an Avaki object such as a *data service* or a *database operation*, or to a table in a relational database. The mapping lets you address each mapped object by the name of the corresponding table in the metadata model. You can also derive a *view model* schema from a metadata model. When you do this, you ensure that the results of any data service deployed from the view model will conform to the metadata model's schema.

**NFS client**

A machine that mounts the Avaki *data catalog* (or a portion of it) as a directory by connecting to an Avaki *data grid access server*. An NFS client need not have Avaki software installed. (NFS—Network File System—lets you add file systems located on a remote computer to the directory structure on your own computer.)



## **ODBC**

ODBC (Open DataBase Connectivity) is an API for databases on Windows. An ODBC driver (such as the the Sybase Organic ODBC driver included with Sybase ASE) allows Avaki to communicate with Windows database applications.

## **on-demand caching**

A scheme by which an object is cached only if it's used—for example, results are cached when a *database operation* or a *data service* is executed, or a file is cached when a user or application reads it. On-demand caching uses a fixed expiration interval to determine data freshness. On-demand caching is suitable for objects that are rarely accessed or that change at irregular intervals. Contrast with *scheduled caching*.

## **pin**

To mark an *Avaki directory* or file for *scheduled caching*. See also *cache service*.

## **plug-in**

See *data service plug-in*.

## **primary GDC**

See *grid domain controller*.

## **proxy server**

An *Avaki server* that allows *Avaki domains* on opposite sides of a firewall or a Network Address Translator (NAT) to communicate with one another.

## **queries**

See *ad-hoc query*.

## **query engine**

An *Avaki service* that executes SQL queries against the *SQL views* (tables) that make up the *Avaki virtual database*. A query engine analyzes queries, pushes as much of the work as possible down to the underlying relational database (if there is one), and performs the remaining operations (such as joins across tables from different databases) itself. There is a query engine on each *grid server*.

## **remote cache**

A *cache service* that runs on a grid server that is remote from an *Avaki service* (a *database operation* or a *data service*) that generates cachable data. The remote cache stores results produced by distant services so the results don't have to be fetched over the network to satisfy every new request. Users and applications that access remote data through the cache may have access to cached copies even when the remote data source is unavailable. See also *local cache*.

**scheduled caching**

A scheme by which an object is cached according to a schedule that you create. The schedule specifies when the object is first cached and how often (or following what trigger event, such as a change to a file) the cache is refreshed. If the object is a *data service* or a *database operation*, the schedule runs it to put fresh results in the cache. Scheduled caching, which overrides other types of caching, is suitable for objects that are updated frequently or on a regular basis. Contrast with *on-demand caching*.

**schedule exclusion**

A named period of time during which scheduled activities can be prevented from running. You can apply an exclusion to as many schedules as you want. Scheduled activities include refreshing *Avaki shares* and imported user accounts, and caching files, directories, and the results of *database operations*, *data services*, and *generated views*.

**secondary GDC**

See *grid domain controller*.

**service**

An Avaki object that performs a function in the domain (stores data or authenticates users, for example). Services provided in Avaki software include *Avaki directories*, *Avaki shares*, *Avaki servers*, *authentication services*, *execution services*, and user accounts.

**share**

A point of connection between the Avaki *data catalog* and a native file system or file system tool. Avaki supports two kinds of shares: *Avaki shares* and *CIFS shares*.

**share server**

An *Avaki server* whose only task is to manage *Avaki shares*—local directories that are exported (shared) into the *data catalog*. (Grid servers can also manage shares.)

**shared directory**

See *Avaki share*.

**soft link**

A pointer to a particular location (name) in the Avaki *data catalog*. If the object at that location is moved, deleted, or renamed, the soft link leads nowhere. Soft links can be created only in the CLI. Contrast with *hard link*.

**SQL view**

A virtual table—a *data catalog* entry that represents a table in a relational database, a *database operation*, or a *data service*. SQL views can be created in three ways:

- Provisioned directly from a table in an underlying database
- Generated from a database operation or data service
- Mapped from a database table, a database operation, or a data service, using the *Avaki Studio* metadata model editor

Every SQL view is part of the Avaki *virtual database*. SQL views are treated as relational tables by the Avaki *query engine*. SQL view data can be accessed using standard SQL statements by connecting to Avaki with ODBC or JDBC, or via an Avaki *virtual database operation*.

### **update notification**

A message issued when a *generated view* is updated. A view that receives data from another view can be configured to regenerate itself (using the new data) upon receipt of an update notification.

### **view generator**

A mechanism that does one of the following: extracts data from a file or an HTTP source, obtains data from an Avaki *data service*, or uses an Avaki *database operation* to extract data from a relational database. The view generator can display the data, perform an XSLT transform, save the data as a *generated view* file, and/or update a database. Contrast with *data service*.

### **view model**

The graphical representation of a data flow that you can build in *Avaki Studio*. A view model typically includes one or more input sources (such as *database operations* or *data services*), one or more operations to combine or transform the data, and an output target. When you deploy a view model, it becomes an Avaki data service.

### **virtual database**

The set of all *SQL views* in an *Avaki domain*, including those provisioned from external databases and those generated from *data services* and *database operations*. You can execute SQL queries on the SQL views in the virtual database as if they were tables in a single database.

### **virtual database operation**

A *database operation* whose source database is the Avaki *virtual database* itself. Use virtual database operations if you want to encapsulate and reuse SQL SELECT queries against *SQL views* (provisioned or generated).

### **web services client**

See *WS client*.

## **WS client**

(Also web services client.) A tool or a piece of code that is part of a customer application and that makes SOAP calls to web services on an Avaki grid server. The SOAP calls can request data from the Avaki *data catalog*, from a *database operation*, or from a *data service*.

---

# Master Index

In electronic copies of this book, the index links to other books in the documentation set work only as long as the PDF files are stored in the same directory.

## Key

**AD:** *Administration Guide*

**API:** *API Guide*

**C:** *Command Reference*

**O:** *Overture*

**P:** *Provisioning & Advanced Data Integration Guide*

**S:** *Data Integration with Avaki Studio*

---

## Symbols

\* asterisks in command syntax AD:xvi, C:xv, P:xi  
- hyphens in command syntax AD:xvi, C:xvi, P:xii  
+ plus signs in command syntax AD:xvi, C:xv, P:xi  
.amm files S:11  
.avm files S:11  
.js files S:11  
.jsi files S:11, S:75  
  sample S:115  
.NET  
  AvakiAPI.disco WSDL discovery file API:3  
  sample web services client API:9  
  SSL certificates API:9  
.project files S:11  
<> angle brackets in command syntax AD:xvi, API:vii, C:xv,  
  P:xi  
= equal signs in command syntax AD:xvi, C:xvi, P:xii  
[ ] square brackets in command syntax AD:xv, C:xv, P:xi  
\_ (underscore) characters in Avaki names API:81  
{ } curly brackets in command syntax AD:xv, C:xv, P:xi  
| vertical bars in command syntax AD:xv, C:xv, P:xi

## A

About My Domain screen AD:98  
AbstractTransformer class P:243  
AbstractTransformerFactory class P:244  
access control in view models S:74  
access control lists, See ACLs  
accessibleDBOp SOAP operation API:42  
accessibleDS SOAP operation API:36  
accessiblePath SOAP operation API:19  
accounts for grid users, See users AD:167  
ACLs  
  about O:45  
  adding users and groups AD:243

## ACLs (*continued*)

  defined AD:349, API:83, C:307, O:61, P:289, S:175  
  deny permissions ineffective for owners, admins O:46  
  displaying AD:237, C:186  
  for database operations P:22, P:36  
  for SQL views P:46  
  granting or denying access to everyone O:44  
  in grid groups O:43  
  interpreting O:48  
  modifying AD:239, C:41, S:97  
  on cached objects O:50  
  on new Avaki shares AD:261  
  on new files O:49  
  ownership AD:242, O:46  
  permissions in AD:242, AD:307  
  removing users and groups from AD:242  
  sample O:45  
  setting for a grid object AD:171  
  setting for database operations P:14  
  using interconnect IDs to add users and groups to AD:304  
Active Directory AD:148  
  domain users group AD:157, AD:159, C:155  
  See also authentication services, LDAP AD:148  
addInputParameter JavaScript method for data service  
  plug-ins P:202  
addInputStream JavaScript method for data service  
  plug-ins P:202  
ad-hoc queries  
  as web services  
    AdHocDBOPExecutionParams complex type API:13  
    executeAdHocDBOp SOAP operation API:43  
    executeAdHocDBOpWithOutput SOAP operation API:44  
    executeAdHocDBOpWithOutputAttach SOAP  
      operation API:46  
    executeAdHocDBOpWithOutputString SOAP  
      operation API:47

---

- ad-hoc queries (*continued*)
  - code samples API:74
  - defined AD:349, API:83, C:307, O:61, P:289, S:175
  - enabling C:66
  - enabling on a database connector P:4
  - executing C:63
  - on virtual database
    - executing C:282
    - parameter types, specifying C:283
    - parameter types, specifying C:65
    - using JDBC driver to run API:69, API:74
- AdHocDBOPExecutionParams complex type API:13
- administrative user accounts, setting up AD:44
- Administrators group O:43
  - about AD:45
  - permissions for AD:240
- admission policies AD:332
  - about AD:85
  - adding C:91
  - creating AD:87
  - deleting AD:88, C:97
  - displaying C:114
  - displaying Windows domains for C:114
  - setting default policies C:112
  - setting Windows domain info C:111
  - unsetting Windows domains for C:116
- aggregate functions S:110
  - in SQL statements, aliasing column names for P:25
- Aggregate operator S:108
- AIX requirements AD:3
- algorithms for join operator S:155
- aliases
  - for column names P:25, S:42
  - aliases for GDC machines AD:10
- Allow permission in ACL AD:243, O:48
- angle brackets in command syntax AD:xvi, API:vii, C:xv, P:xi
- Apache Ant for compiling data service plug-ins P:184
- Apache Axis API:5
  - data catalog example API:18
  - data service example API:35
  - database operations example API:42
- APIs
  - data catalog API:18
  - data services API:34
  - database operations API:40
  - for data service plug-ins
    - about P:185
    - distributed transaction API P:188
    - general data service API P:186
  - TrAX (Transformation API for XML) P:243
  - web services
    - about API:2
    - data service API:35
    - reference API:11
- AROMValue parameters P:212
- As is permission in ACL AD:242
- ASE, see Sybase ASE
- asterisks in command syntax AD:xvi, C:xv, P:xi
- attribute --delete command C:19
- attribute --list command C:19
- attribute --update command C:21
- attributes
  - about AD:245
  - configuring for SQL views P:44
  - creating AD:248
    - and modifying S:100
  - defined AD:349, API:83, C:307, O:61, P:289, S:175
  - deleting AD:254, C:19
  - displaying AD:246
  - displaying details about C:19
  - ldap/importOnDemand AD:158
  - nis/importOnDemand AD:164
  - of cache services C:291
  - of grid servers C:290
  - of patches C:290
  - searching on AD:233
  - setting values AD:252, C:21
  - system attributes AD:248
  - types of AD:250, C:22, S:101
  - user-defined attributes AD:248
  - who can edit S:101
- audit logging
  - about O:14
  - configuring AD:319
  - events captured by AD:322
- audit logs AD:108
- authentication in Avaki O:41
  - using AvakiPrincipal API:13
- authentication services
  - configuring default groups C:218
  - configuring default users C:220
  - configuring GIDs C:217, C:221
  - configuring UIDs C:219, C:222
  - defined AD:349, API:83, C:307, O:61, P:289, S:175
  - deleting AD:166
  - displaying information about AD:166
  - grid, creating groups on C:141
- LDAP
  - adding schedule exclusions for refreshing C:152
  - adding search bases C:160
  - deleting authentication services C:153
  - deleting import schedules C:154
  - deleting search bases C:160
  - displaying information about C:157
  - enabling users C:250
  - importing groups from AD:159, C:155
  - importing users from AD:157, C:155
  - integrating into the grid AD:148, C:157
  - listing import schedules C:158
  - scheduling refreshes AD:185
  - scheduling user imports C:149
  - setting page size for imports AD:145
  - updating C:160

authentication services (*continued*)

- NIS
  - adding schedule exclusions for refreshing C:180
  - deleting authentication services C:181
  - deleting import schedules C:181
  - displaying info about C:183
  - enabling users C:250
  - importing groups from AD:165, C:182
  - importing users from AD:164, C:182
  - integrating into the grid AD:162, C:184
  - listing import schedules C:184
  - scheduling user imports C:177
  - updating C:185
- refreshing imported accounts AD:185
- specifying for JDBC connections API:69
- types O:41
  - specifying for JDBC connections API:69
- authentication using AvakiPrincipal API:9
- auto-restart
  - about AD:37, C:5
  - configuring for a DGAS C:5
  - configuring for a GDC AD:38, C:9
  - configuring for a grid server AD:51, C:9
  - configuring for a proxy server C:12
  - configuring for a share server C:15
- avaki attribute --delete C:19
- avaki attribute --list C:19
- avaki attribute --update C:21
- avaki backup C:23
- avaki cache --evict C:24, C:32
- avaki cache --evict --all C:25
- avaki cache --evict --deleted C:26
- avaki cache --get C:27
- avaki cache --invalidate C:27
- avaki cache --invalidate --all C:28
- avaki cache --invalidate-dataservice-results C:29
- avaki cache --invalidate-dbop-results C:30
- avaki cache --list C:31
- avaki cache --set C:33
- avaki cache --unset C:34
- avaki cat C:35
- avaki categories --add-to-category C:35
- avaki categories --create C:36
- avaki categories --delete C:37
- avaki categories --describe C:37
- avaki categories --list C:38
- avaki categories --remove-from-category C:38, C:40
- avaki categories --set-description C:39
- avaki cd C:41
- avaki chmod C:41
- avaki chown C:42
- avaki client C:45
- avaki client --connect command AD:94
- avaki cp C:46
- avaki database operation --list-schedules C:88
- avaki dataservice --add-schedule C:48
- avaki dataservice --create C:52
- avaki dataservice --delete C:52
- avaki dataservice --delete-schedule C:54
- avaki dataservice --depends C:54
- avaki dataservice --execute C:55
- avaki dataservice --generate-sql view C:56
- avaki dataservice --info C:57
- avaki dataservice --list-schedules C:58
- avaki dataservice --update C:58
- avaki dbconn --allow-dbop-creation C:59
- avaki dbconn --delete C:61
- avaki dbconn --disallow-dbop-creation C:62
- avaki dbconn --execute C:63
- avaki dbconn --info C:64
- avaki dbconn --jdbc C:66
- avaki dbconn --provision-tables C:71
- avaki dbconn --show-tables C:73
- avaki dbconn --test C:72
- avaki dbop --add-schedule C:73
- avaki dbop --delete C:78
- avaki dbop --delete-schedule C:78
- avaki dbop --depends C:79
- avaki dbop --execute C:80
- avaki dbop --generate-sql view C:82
- avaki dbop --info C:83
- avaki dbop --jdbc C:83
- avaki dbop --jdbc --create-virtual-dbop C:87
- avaki dgas --add-admission-policy C:91
- avaki dgas --add-group-mapping C:92
- avaki dgas --add-user-mapping C:94
- avaki dgas --cifs-share-info C:95
- avaki dgas --clear-cached-credentials C:95
- avaki dgas --create-cifs-share C:96
- avaki dgas --delete-admission-policy C:97
- avaki dgas --delete-cache C:98
- avaki dgas --delete-cifs-share C:99
- avaki dgas --delete-group-mapping C:99
- avaki dgas --delete-user-mapping C:100
- avaki dgas --disconnect-cifs-client C:101
- avaki dgas --get-cache-size C:101
- avaki dgas --get-cache-statistics C:102
- avaki dgas --get-free-disk-space C:102
- avaki dgas --get-properties C:103
- avaki dgas --get-property C:103
- avaki dgas --get-property-list C:104
- avaki dgas --initialize C:104
- avaki dgas --list-cifs-clients C:105
- avaki dgas --list-cifs-shares C:105
- avaki dgas --list-group-mappings C:106
- avaki dgas --list-user-mappings C:106
- avaki dgas --modify-cifs-share C:107
- avaki dgas --read-log-properties C:107
- avaki dgas --reset-cache-statistics C:108
- avaki dgas --save-cache C:109
- avaki dgas --self-map C:109
- avaki dgas --self-unmap C:111
- avaki dgas --set-admission-policy-domain C:111
- avaki dgas --set-default-admission-policy C:112
- avaki dgas --set-property C:113

- avaki dgas --show-admission-policies C:114
- avaki dgas --show-admission-policy-domain C:114
- avaki dgas --sync-cache C:115
- avaki dgas --unset-admission-policy-domain C:116
- avaki dgas --unset-property C:117
- Avaki directories, See directories, Avaki
- avaki directory --add-schedule C:117
- avaki directory --cache C:122
- avaki directory --delete-schedule C:122
- avaki directory --do-not-cache C:126
- avaki directory --list-schedules C:126
- avaki domain --create C:127
- avaki domain --disconnect C:127
- avaki domain --info C:128
- avaki domain --interconnect C:128
- Avaki domains, See domains, Avaki
- Avaki EII software
  - overview O:1
  - typical deployment O:17
- avaki executionservice --info C:129
- avaki executionservice --set C:129
- avaki file --add-schedule C:130
- avaki file --cache-on-demand C:134
- avaki file --delete-schedule C:135
- avaki file --do-not-cache C:136
- avaki file --list-schedules C:136
- avaki file --pin C:137
- Avaki functions S:73
- Avaki Functions menu S:74
- avaki group --add --user C:138
- avaki group --create C:141
- avaki group --delete C:143
- avaki group --delete --user C:144
- avaki group --info C:145
- avaki group --list-user C:147
- avaki help C:148
- avaki id C:149
- Avaki installation directory AD:350, API:84, C:308, O:62, P:290, S:176
- avaki ldap --add-schedule C:149
- avaki ldap --delete C:153
- avaki ldap --delete-schedule C:154
- avaki ldap --import C:155
- avaki ldap --info C:157
- avaki ldap --integrate C:157
- avaki ldap --list-schedules C:158
- avaki ldap --searchbase C:160
- avaki ldap --update C:160
- avaki ln C:161
- avaki locks --clear C:163
- avaki locks --list C:164
- avaki login C:164
- avaki logout C:165
- avaki ls C:166
- avaki mkdir C:167
- avaki monitor --add C:167
- avaki monitor --clear C:168
- avaki monitor --create C:170
- avaki monitor --delete C:171
- avaki monitor --list C:172
- avaki monitor --result C:172
- avaki monitor --start C:173
- avaki monitor --stop C:174
- avaki mv command C:176
- avaki nis --add-schedule C:177
- avaki nis --delete C:181
- avaki nis --delete-schedule C:181
- avaki nis --import C:182
- avaki nis --info C:183
- avaki nis --integrate C:184
- avaki nis --list-schedules C:184
- avaki nis --update C:185
- avaki passwd C:185
- avaki permissions C:186
- Avaki perspective in Studio S:13
- avaki plugin command P:184
- avaki plugin --generate C:187
- avaki proxy --add C:191
- avaki proxy --delete C:191
- avaki proxy --list C:192
- avaki pwd C:193
- avaki replica --add C:193
- avaki replica --config command C:193
- avaki replica --delete C:194
- avaki replica --info C:194
- avaki replicate --synch C:195
- avaki rm C:195
- Avaki rowset XML
  - class-name element P:279
  - column-display-size element P:279
  - column-index element P:279
  - core schema P:277
  - rowset-specific schema P:279
  - sample schema P:280
  - schema overview P:277
- avaki schedule --delete C:196
- avaki schedule --info C:197
- avaki schedule --list C:197
- avaki schedule --print-iterations C:198
- avaki scheduleexclusion --create --custom C:198
- avaki scheduleexclusion --create --daily C:199
- avaki scheduleexclusion --create --monthly C:201
- avaki scheduleexclusion --create --weekly C:203
- avaki scheduleexclusion --create --yearly C:205
- avaki scheduleexclusion --delete C:207
- avaki scheduleexclusion --info C:208
- avaki scheduleexclusion --list C:209
- avaki search (execute) C:211
- avaki search --create command C:209
- avaki search --delete C:210
- avaki search --get-rehash-level C:212
- avaki search --info C:214
- avaki search --rehash C:215
- avaki search --set-rehash-level C:215
- avaki security --config C:216
- avaki security --default-gid C:217



- avaki security --default-group C:218
- avaki security --default-uid C:219
- avaki security --default-user C:220
- avaki security --gid C:221
- avaki security --info C:222
- avaki security --uid C:222
- avaki server --dgas --connect C:223
- avaki server --dgas --destroy C:224
- avaki server --dgas --stop C:225
- avaki server --grid --connect C:225
- avaki server --grid --destroy C:226
- avaki server --grid --stop C:227
- avaki server --proxy C:228
- avaki server --share --connect C:228
- avaki server --share --disconnect C:229
- avaki server --share --stop C:230
- Avaki servers
  - distribution of data catalog among O:38
  - hardware and operating system requirements for O:16
  - qualified names for O:32
- avaki share --add-rehash-schedule C:231
- avaki share --add-share-servers C:232
- avaki share --create C:235
- avaki share --delete-rehash-schedule C:236
- avaki share --disconnect C:238
- avaki share --get-local-path C:238
- avaki share --get-status C:239
- avaki share --list-rehash-schedules C:239
- avaki share --list-share-servers C:240
- avaki share --rehash C:240
- avaki share --remove-share-servers C:241
- avaki share --set-local-path C:241
- avaki share --set-share-servers C:242
- avaki share --set-status C:243
- avaki share --update-share-servers C:244
- Avaki shares
  - about AD:257
  - adding schedule exclusions for rehashes C:234
  - adding share servers C:232
  - behavior during failover AD:112
  - bringing on line AD:286
  - changing configuration of AD:266
  - changing encryption levels AD:279
  - changing permissions AD:239
  - changing the owner AD:261
  - configuring exclusions for refresh schedules AD:274
  - copying into, out of, and within AD:213
  - creating AD:258, C:235
  - defined AD:350, API:84, C:308, O:62, P:290, S:176
  - deleting C:195
  - disconnecting C:238
  - disconnecting permanently AD:287
  - forcing refresh AD:262
  - icon for O:29
  - linking AD:217
  - local paths for, obtaining C:238
  - modifying load balance factor C:244
  - moving AD:210
- Avaki shares (*continued*)
  - moving source directories AD:283
  - naming of files and directories in AD:207
  - online status, setting C:243
  - organizing O:37
  - permissions on new files in O:49
  - refresh schedules
    - adding C:231
    - deleting C:236
    - listing C:239
  - refresh schedules for AD:266
  - refreshing C:240
  - removing entries from refresh schedules AD:278
  - removing share servers from AD:265
  - renaming AD:212
  - setting load balancing factor AD:280
  - setting local paths C:241
  - setting names AD:260
  - share servers
    - listing C:240
    - removing C:241
    - replacing C:242
  - shutting down AD:287, C:238
  - status, displaying C:239
  - taking off line AD:285
  - uploading files to AD:282
  - with multiple share servers AD:263
  - write access and user accounts AD:12
  - See also share servers
- avaki shell C:245
- avaki sql view --delete C:246
- avaki sql view --get-description C:246
- avaki sql view --set-description C:247
- avaki status C:248
- Avaki Studio
  - about O:9, O:17, S:1
  - Avaki perspective, about S:16
  - defined AD:350, API:84, C:308, O:62, P:290, S:176
  - getting started S:9
  - installing in Windows AD:24
  - limitations of data services created in P:78, S:3
  - log properties file for AD:317
  - metadata models, See metadata models
  - operators S:5, S:107
  - projects, creating S:13
  - requirements for running AD:3
  - setting system properties for AD:129
  - starting S:9
  - time required to upgrade AD:341
  - view models
    - about S:2
    - configuring input sources S:43
    - creating S:29
    - deploying as data services S:50
    - sample workflow for S:29
    - testing S:49
  - workflow S:25

avaki upgrade C:249  
 avaki upgrade --info C:250  
 avaki user C:250  
 avaki user --create C:251  
 avaki user --db-mapping --add C:252  
 avaki user --db-mapping --delete C:253  
 avaki user --db-mapping --list C:255  
 avaki user --delete C:257  
 avaki user --info C:258  
 avaki user --list-group C:258  
 avaki view --add-schedule C:259  
 avaki view --create --database C:263  
 avaki view --create --data-service C:266  
 avaki view --create --file C:267  
 avaki view --delete C:272  
 avaki view --delete-schedule C:272  
 avaki view --depends C:272  
 avaki view --garbage-collect C:273  
 avaki view --info C:274  
 avaki view --list-schedules C:274  
 avaki view --regenerate C:273  
 avaki view --set-property C:275  
 avaki view --update C:279  
 avaki virtualdatabase --allow-dbop-creation C:280  
 avaki virtualdatabase --disallow-dbop-creation C:281  
 avaki virtualdatabase --execute C:282  
 avaki virtualdatabase --show-tables C:283  
 avaki virtualeschema --deploy C:285  
 avaki virtualeschema --undeploy C:286  
 avaki whoami C:286  
 Avaki\_JDBCStandAlone.jar file API:66  
 Avaki\_JDBCStandAlone\_Minus3rd.jar file API:66  
 AvakiAPI.disco file API:3  
 AvakiAPIDocLit.wsdli file API:3  
 AvakiAPIRpcEnc.wsdli file API:3  
 AvakiAPIWithMIMEDocLit.wsdli file API:3  
 AvakiAPIWithMIMERpcEnc.wsdli file API:3  
 avakijdbc.properties file API:67  
 AvakiPrincipal complex type API:13  
 Axis, See Apache Axis

## B

backup command C:23  
 backups on Avaki servers AD:113  
 batch mode, JDBC API:77  
     configuring database operations for P:27, P:250  
 bindings.xml file  
     copying during upgrade AD:344  
     on grid servers AD:50  
     on proxy servers AD:300  
     on share servers AD:59  
 block size file attribute C:290  
 blocks file attribute C:290  
 bootstrapping  
     in Unix AD:16  
     in Windows AD:23  
 brackets, See curly brackets, square brackets, angle brackets

browsers, See web browsers  
 build.xml file for data service plug-ins P:184, P:197  
 BusinessObjects software unable browse Avaki objects with  
     underscores API:81

## C

cache --evict --all command C:25  
 cache --evict command C:24, C:32  
 cache --evict --deleted command C:26  
 cache --get command C:27  
 cache --invalidate --all command C:28  
 cache --invalidate command C:27  
 cache --invalidate-dataservice-results command C:29  
 cache --invalidate-dbop-results command C:30  
 cache --list command C:31  
 cache services  
     about P:119  
     associating with data grid access servers P:113  
     associating with grid servers P:111  
     coherence windows P:107  
     configuring P:116  
     configuring per file P:117  
     defined AD:351, API:85, C:309, O:63, P:291, S:177  
     disassociating from data grid access servers P:114  
     disassociating from grid servers P:112  
     evicting cached files and directories P:135  
     invalidating cached items P:136  
     listing P:116  
     listing cached data services P:163  
     listing cached database operations P:148  
     listing cached virtual database operations P:148  
     listing pinned files and directories P:134  
     on-demand caching P:119  
     on-demand caching of database operation and data service  
         results P:108  
     on-demand caching of files P:107  
     overriding default settings P:117  
     pinning data services P:152  
     pinning database operations P:139  
     pinning files and directories P:120  
     pinning virtual database operations P:139  
     scheduled caching P:119  
     scheduled caching of database operation and data service  
         results P:109  
     scheduled caching of files P:107  
     tagging files and directories P:129  
     unmarking cached items P:135  
     unscheduling cached files and directories P:135  
     viewing details about P:116  
     See also caches and caching  
 cache --set command C:33  
 cache --unset command C:34  
 caches  
     adding schedules for data services C:48  
     adding schedules for database operations C:73  
     adding schedules for directories C:117  
     adding schedules for files C:130

caches (*continued*)

- bad port, properties for AD:141
- configuring associated server or user C:33
- configuring threads for AD:131
- data service plug-in, properties for AD:137
- deleting schedules for data services C:54
- deleting schedules for database operations C:78
- deleting schedules for directories C:122
- deleting schedules for files C:135
- DGAS
  - clearing user credentials AD:117
  - clearing user credentials from C:95
  - configuring block size for reads AD:81
  - configuring frags per block for reads AD:81
  - configuring location of AD:73
  - controlling cache size AD:124
  - deleting files and directories AD:119
  - deleting objects from C:98
  - displaying cache statistics C:102
  - displaying current size C:101
  - displaying free disk space on cache machine C:102
  - forcing a refresh AD:121
  - managing AD:117
  - mapping cache AD:336
  - resetting statistics C:108
  - saving a copy C:109
  - saving copies AD:120
  - setting remote caches for AD:90
  - syncing AD:121
  - viewing and resetting statistics AD:123
  - warming and updating C:115
- displaying associated server or user C:27
- displaying tagging information C:31
- for tables in virtual database, property for AD:144
- listing schedules for data services C:58
- listing schedules for database operations C:88
- listing schedules for directories C:126
- listing schedules for files C:136
- local S:182
- local and remote API:63
- marking directories for no caching C:126
- marking files for no caching C:136
- remote object stub, properties for AD:144
- schedule exclusion, properties for AD:144
- scheduled caching S:184
- settable attributes of C:291
- setting invalidate queue for AD:135
- setting local directory for AD:135
- setting remote caches for command clients AD:95
- uncoupling associated server or user C:34

See also cache services and caching

## caching

- about O:13
- and JDBC programs O:55
- benefits to performance O:54
- configuring ACLs for O:50

caching (*continued*)

- configuring Avaki clients for O:55
- data service results P:108, S:51
  - tagging for on-demand caching P:159
- database operations P:108
- defined AD:351, API:85, C:309, O:63, P:291, S:177
- files O:56, P:107
- JDBC and caching of database operation results API:62
- local AD:356, API:90, C:314, O:14, O:53, O:68
- local vs. remote O:59
- local, defined P:296
- of data service results O:57
- of database operation results O:57
- on DGAS O:54
- on-demand AD:357, API:91, C:315, O:69, P:296, S:183
- permissions and access control O:59
- remote O:14, O:53
  - defined AD:357, API:91, C:315, O:69, P:297, S:184
- scheduled AD:357, API:91, C:315, O:69
- scheduled, defined P:297
- turning off for specified files and directories P:132
- See also caches and cache services
- callable statements API:72, API:73
- case sensitivity in Avaki naming AD:206
- cat command C:35
- catalog browser S:18
- categories
  - about AD:221
  - adding objects to AD:226, C:35, S:105
  - adding SQL views P:47
  - and permissions AD:222–AD:223
  - browsing AD:222
  - contents of S:18
  - creating AD:224, C:36
  - default, contents of S:18
  - defined AD:351, API:85, C:309, O:63, P:291, S:177
  - deleting AD:230, C:37, S:106
  - displaying S:104
  - for logging AD:318
  - icon for O:29
  - listing categories in domain C:38
  - managing S:103
  - permissions in O:48
  - removing objects from AD:228, C:38, C:40, S:106
  - setting descriptions for C:39
  - showing descriptions C:37
  - using to organize data O:36
  - using to solve access problems O:49
- categories --add-to-category command C:35
- categories --create command C:36
- categories --delete command C:37
- categories --describe command C:37
- Categories directory O:35
- categories --list command C:38
- categories --remove-from-category command C:38, C:40
- categories --set-description command C:39
- cd command C:41
- certificates, SSL, See SSL certificates

- change time file attribute C:290
- characters
  - in column aliases in database operations S:42
  - in command syntax AD:xv, C:xv
  - in cron schedules C:298
  - in domain names, restrictions on AD:41
  - in JavaScript identifiers S:42
  - in metadata model names, restrictions on S:91
  - in names of Avaki objects, restrictions on AD:207
  - wildcards in searches AD:235
- CHARSET JDBC property for ASE and IQ AD:7
- chmod command C:41
- chmod SOAP operation API:19
- chown command C:42
- chown SOAP operation API:20
- chunk size for sorting, controlling AD:139, S:76
- CIFS
  - accessing data grid through AD:338
  - releasing CIFS ports on a DGAS AD:66
- CIFS clients
  - defined AD:351, API:85, C:309, O:63, P:291, S:177
  - disconnecting C:101
  - displaying connected clients C:105
  - requirements for O:17
  - setting up AD:93
- CIFS shares
  - accessing AD:203
  - creating AD:125, C:96
  - defined AD:351, API:85, C:309, O:63, P:291, S:177
  - deleting C:99
  - displaying connected clients C:105
  - displaying information about C:95
  - listing C:105
  - managing AD:125
  - mapping to a network drive AD:204
  - modifying C:107
- class element P:261
- class-name element P:279
- classpath, configuring for JDBC drivers API:67
- client attribute caching AD:336
- client command C:45
- client system properties AD:128
- clients
  - about O:17
  - hardware and operating system requirements for O:16
  - message timeout properties for AD:133
  - setting size of write invalidation queue of cache for AD:136
  - setting system properties for AD:129
  - See also Avaki Studio, CIFS clients, command clients, NFS clients, web clients, WS clients
- code samples
  - ad-hoc queries API:74
  - data catalog API API:18
  - data services API API:35
  - database operations API API:42
  - Java data service plug-ins P:190
  - JDBC batch mode API:77
  - using JDBC drivers API:77
- coherence window cache attribute C:291
- coherence window property, remote AD:141
- coherence windows for caching P:107
- coherenceWindow element P:261
- colors in Studio display, setting S:23
- column-display-size element P:279
- column-index element P:279
- columns
  - aliasing P:25
  - combining with Projection operator S:46
  - from input elements, menus of S:71
  - from input result sets, accessing S:68
  - name property S:60
  - precision property S:61
  - scale property S:61
  - type property S:61
- com.avaki.badPortCacheSize system property AD:141
- com.avaki.badPortExpiration system property AD:141
- com.avaki.cache.cacheDir system property AD:135
- com.avaki.cache.maxReaderThreads system property AD:131
- com.avaki.cache.writeInvalidationQueueSize system property AD:136
- com.avaki.content.encryptionLevel system property AD:139
- com.avaki.dataservice.pluginCacheSize system property AD:137
- com.avaki.dataservice.styleSheetCachePoolSize system property AD:137
- com.avaki.dataservice.styleSheetCacheSize system property AD:137
- com.avaki.DBOProtocolSoTimeout system property AD:134
- com.avaki.generatedXMLIndentSize AD:142
- com.avaki.HttpPort system property AD:140
- com.avaki.HttpsPort system property AD:140
- com.avaki.jobStatusExpiration system property AD:145
- com.avaki.lasInvoker.cacheSize system property AD:144
- com.avaki.lasInvoker.poolSize system property AD:144
- com.avaki.ldap.resultPageSize system property AD:145
- com.avaki.maxActiveCachables system property AD:136
- com.avaki.mux.channelSoTimeout system property AD:135
- com.avaki.mux.connectTimeout system property AD:134
- com.avaki.mux.maxParallelChannels system property AD:142
- com.avaki.mux.maxWriteChunk system property AD:142
- com.avaki.mux.sendBufferSize system property AD:143
- com.avaki.proxy.retryDelay system property AD:133
- com.avaki.proxy.retryTimeout system property AD:133
- com.avaki.proxyIOProtocolSoTimeout system property AD:134
- com.avaki.proxyKeepAliveParams system property AD:140
- com.avaki.queryEngine.sortChunkSize AD:139
- com.avaki.remoteconfig.coherenceWindow system property AD:141
- com.avaki.result.gcInterval system property AD:136
- com.avaki.retryDelay system property AD:133
- com.avaki.retryTimeout system property AD:133
- com.avaki.rmiRegistrySoTimeout system property AD:134
- com.avaki.rpcTimeout system property AD:134
- com.avaki.scheduleExclusionCacheExpiration system property AD:145

- com.avaki.scheduleExclusionCacheSize system property AD:145
- com.avaki.shareIOProtocolSoTimeout system property AD:134
- com.avaki.shareReadBufferSize system property AD:138
- com.avaki.shareReadbufPoolSize system property AD:138
- com.avaki.shareServerCircularLinkChecking system property AD:138
- com.avaki.shareServerThreadPoolSize system property AD:138
- com.avaki.vaultStateCacheSize system property AD:137
- com.avaki.VirtualDbTableCacheSize system property AD:144
- com.sybase.avaki.tdsPort system property AD:50, AD:145, API:71
- command clients
  - connecting C:45
  - defined AD:352, API:86, C:310, O:64, P:292, S:178
  - disconnecting C:45
  - installing in Windows AD:24
  - installing on Unix AD:18
  - obtaining information about C:45
  - setting up AD:94
- commands
  - listing C:148
  - syntax conventions for AD:xv, C:xiv, P:x
  - viewing online usage information C:148
- compatibility properties, setting for Windows 2003 AD:22
- complex types API:12
  - AdHocDBOPExecutionParams API:13
  - AvakiPrincipal API:13
  - DataCatalogAttribute API:14
  - DataCatalogEntry API:15
  - DataCatalogPermission API:15
  - DataServiceExecutionParams API:16
  - DBOPExecutionParams API:16
  - SearchQuery API:17
  - SearchResult API:17
- condition field for Iterator operators S:151
- connect ports
  - default AD:6, AD:9, AD:10
  - defined AD:352, API:86, C:310, O:64, P:292, S:178
  - for DGAS C:224
    - changing AD:74
  - for GDCs C:127
    - changing AD:50
  - for grid servers C:226
    - changing AD:50
  - for proxy servers C:228
    - changing AD:300
  - for share servers C:229
    - changing AD:60
- connectinfo.txt file AD:131
- connection pooling S:36
- connection properties
  - for JDBC drivers API:68
  - for XA drivers C:70, P:7, S:37
- connection strings
  - for databases AD:3
  - for JDBC drivers API:71

- connections in view models, creating S:57
- console view S:22, S:50
- conventions
  - for command syntax C:xiv
  - for commands AD:xv
  - for screen examples AD:xv, API:vi, C:xv, P:xi
- cp command C:46
- cron expressions in schedules AD:185, AD:267, AD:273
- cron schedules
  - configuring C:297
  - values for C:298
- cross-domain messaging
  - disabling AD:313
  - enabling AD:311
- curly brackets in command syntax AD:xv, C:xv, P:xi
- CurrentUser functions S:74
- Custom operator S:111
  - example S:114
- custom types API:12

## D

- data access O:11
  - using WS API API:2
- data catalog
  - about O:27
  - defined AD:352, API:86, C:310, O:64, P:292, S:178
  - distribution among Avaki servers O:38
  - names of objects in O:24
  - organizing O:33
    - Avaki shares O:37
      - using categories O:36
      - using links O:36
  - ownership of objects in O:46
  - top-level directories O:32
  - types of entries O:6
- data catalog API API:18
- data catalog SOAP operations API:18
  - accessiblePath API:19
  - chmod API:19
  - chown API:20
  - fileRead API:21
  - fileReadAttach API:21
  - fileReadString API:22
  - fileWrite API:23
  - getAttributes API:23
  - getSystemAttributes API:24
  - getUserAttributes API:24
  - listDomains API:25
  - listSearches API:25
  - ls API:26
  - lsSize API:26
  - mkdir API:27
  - mkdirParents API:27
  - mkdirParentsServer API:28
  - mkdirServer API:29
  - mv API:29
  - permissions API:30

- data catalog SOAP operations (*continued*)
  - removeAttribute API:31
  - rm API:31
  - search API:32
  - setAttribute API:32
  - tester API:33
  - whoami API:33
- data catalog view S:18
- data expiration intervals P:108, S:51
- data grid access servers
  - associating with cache services P:113
  - disabling auto-restart C:8
  - disassociating from cache services P:114
  - enabling auto-restart C:5
  - registering C:5
  - starting C:5, C:6
  - stopping C:7, C:8
  - unregistering C:8
  - See also DGAS
- data grids
  - about O:1
  - defined AD:355, API:89, C:313, O:67, P:294, S:181
  - typical deployment O:17
- data integration O:21, O:23
- data integrity and HTTPS API:8
- data representation O:11
- data security O:10
- data service plug-ins
  - about P:76, P:175
  - addInputParameter JavaScript method P:202
  - addInputStream JavaScript method P:202
  - build.xml file P:184, P:197
  - choice of Java, JavaScript, or XSLT P:176
  - closing streams P:186
  - command for generating C:187
  - configuring P:81
  - creating in Java with the Plug-in Wizard P:183
  - creating in JavaScript P:200
  - creating in XSLT P:180
  - DbopGroupWorkUnit class P:189
  - DbopPipeWorkUnit class P:190
  - defined AD:352, API:86, C:310, O:64, P:292, S:179
  - examples
    - DBOP and CSV merge Java plug-in P:193
    - distributed transaction Java plug-in P:191
    - rowset input and output Java plug-in P:192
  - Execute JavaScript function P:203
  - input sources and output streams P:177
  - InputSource interface P:186
  - JAR files for P:180
  - logging errors P:196
  - manifest files for P:180, P:197
  - modularity and reusability of P:175
  - parameters
    - about P:178
    - specifying for Java plug-ins C:188
    - specifying for XSLT plug-ins P:181
- data service plug-ins (*continued*)
  - ParameterSpec interface P:187
  - Plugin interface P:186
  - prerequisites for writing in Java P:183
  - relationship to .js files in Studio S:11
  - RowSetFactory class P:188
  - setOutputStream JavaScript method P:202
  - StreamingRowSet interface P:187
  - using Java classes and interfaces in JavaScript plug-ins P:200
  - when to use P:78, S:3
  - XAWorkHandler class P:189
  - XAWorkUnit interface P:189
- data service XML schema
  - class element P:261
  - coherenceWindow element P:261
  - dataService element P:262
  - description element P:263
  - initParameter element P:263
  - inputParameter element P:264
  - inputSource element P:265
  - inputStream element P:266
  - isList element P:266
  - jarurl element P:267
  - logicBox element P:268
  - name element P:269
  - outputStream element P:269
  - ref element P:270
  - target element P:270
  - type element P:270
  - urlLogicBox element P:271
  - value element P:272
  - values element P:272
- data services
  - about O:8, O:23, P:49, P:74
  - adding schedule exclusions C:51
  - and distributed transactions P:78
  - caching of results O:57, P:77
    - permissions O:50
  - caching results S:51
  - calling via JDBC API:72
  - components of P:76
  - configuring caching P:108
  - created in Avaki Studio, limitations of P:78, S:3
  - creating C:52, P:80, P:207
  - defined AD:352, API:86, C:310, O:64, P:292, S:178
  - deleted, purging from cache C:26
  - deleting schedules C:54
  - dependencies for S:22
  - deploying from view models in Avaki Studio S:50
  - displaying dependency lists C:54
  - displaying information about C:57, S:20
  - displaying status of C:248
  - evicting from cache P:164
  - execution services, configuring AD:109
  - generating schema for C:55, P:98
  - generating SQL views from C:56, P:100
  - importing descriptors P:92
  - input parameters, configuring P:84

- data services (*continued*)
  - input streams, configuring P:87
  - invalidating all in cache C:28
  - invalidating one in cache C:27
  - invalidating results in cache C:29
  - listing P:93
  - listing caching schedules for C:58
  - listing in cache P:163
  - location in categories S:18
  - marking for scheduled caching P:152
  - modifying C:58
  - modifying permissions AD:239
  - modifying settings P:94
  - names in data catalog O:24
  - nesting operations S:149
  - output streams, configuring P:86
  - provisioning web services as P:205
  - purging all from cache and unspooling C:25
  - purging one from cache and unspinning C:24
  - qualified names for O:31
  - refreshing cached results C:32
  - removing P:103
  - rowsets as input of P:275
  - rowsets as output of P:274
  - running C:55
  - sample workflow for S:29
  - scheduling for caching C:48
  - schema P:257
  - searching for AD:233
  - setting cache sizes for plug-ins AD:137
  - setting up to run distributed transactions P:80
  - specifying grid servers P:213
  - specifying input parameters P:207
  - specifying input streams P:208
  - specifying output streams P:208
  - specifying plug-ins P:207
  - tagging for on-demand caching P:159
  - testing P:102, P:214
  - unscheduling P:164
  - using for distributed transactions O:25
  - viewing P:98
  - viewing dependencies P:97
  - writing your own vs. using Avaki Studio O:24
  - See also data service plug-ins
  - See also view models
- data services API API:34
- data services SOAP operations API:34
  - accessibleDS API:36
  - executeDS API:36
  - getDSOutput API:37
  - getDSOutputAttach API:38
  - getDSOutputString API:38
  - getDSParameters API:39
  - isDSAvakiXML API:40
  - listDSs API:40
- data structures, SOAP complex types API:12
- data type mappings for SQL views P:39
- data types
  - for JDBC API:76
  - mapping
    - about type mapping files C:301
    - command to specify mapping file C:68
    - format of type mapping files C:301
    - inconsistencies C:302
    - logging of mapping decisions C:304
    - setting source data type C:302
  - specifying for ad-hoc query parameters C:65
  - specifying for database operation parameters C:85
  - specifying for parameters for ad-hoc queries on the virtual database C:283
  - specifying for virtual database operation parameters C:88
  - See also type
- database connectors
  - about O:22, P:3
  - adding groups P:16
  - adding users P:15
  - configuring advanced settings P:247
  - configuring JDBC driver JAR file path P:247
  - configuring permissions C:59, C:62
  - creating C:66, P:3, S:31
  - data type mappings for P:39
  - defined AD:353, API:87, C:311, O:65, P:293, S:179
  - deleting C:61
  - displaying information about C:64
  - displaying SQL views provisioned from C:73
  - editing S:38
  - executing ad-hoc queries C:63, C:66
  - finding in catalog S:38
  - getting information about through JDBC API:75
  - JDBC fetch size P:5
  - location in categories S:18
  - managing SQL views P:20
  - modifying C:66, P:8
  - provisioning SQL views from C:71
  - removing P:21
  - removing groups P:18
  - removing users P:18
  - searching for AD:233
  - setting JDBC fetch size S:36
  - testing C:72, P:19
  - viewing P:8
  - viewing associated database operations P:13
- database drivers
  - copying during upgrade AD:341
  - tested with Avaki AD:3
- database identity mappings P:6, S:36
  - about AD:176
  - adding C:252
  - deleting AD:183, C:253
  - displaying AD:180
  - listing C:255
  - modifying AD:182
  - setting up AD:177
- database operation --list-schedules command C:88

database operation SOAP operations API:40

database operations

about O:7, O:22, P:1

access permissions P:22

adding schedule exclusions C:76

allowing groups to create P:16

allowing users to create P:15

caching of results API:62, O:57

permissions O:50

calling with JDBC API:73

calling with ODBC, JDBC, or SOAP P:38

configuring advanced settings P:247

configuring batch mode settings P:250

configuring caching P:108

configuring permissions C:59, C:62

configuring SQL calls P:251

configuring timeouts P:253

creating C:83, P:22, S:38

defined AD:353, API:87, C:311, O:65, P:293, S:179

deleted, purging from cache C:26

deleting C:78

deleting schedules C:78

dependencies for S:22

displaying dependency lists C:79

displaying information about C:83, S:20

displaying status of C:248

evicting from cache P:150

executing P:36

exposing results as SQL view P:34

generating schema for C:80, P:31

generating SQL views from C:82

invalidating all in cache C:28

invalidating one in cache C:27

invalidating results in cache C:30

listing caching schedules for C:88

listing in cache P:148

location in categories S:18

managing P:21

managing metadata P:30

marking for scheduled caching P:139

modifying C:83, P:28

modifying permissions AD:239

names in data catalog O:24

parameter types, specifying C:85, C:88

preventing groups from creating P:18

preventing users from creating P:18

purging all from cache and unscheduling C:25

purging one from cache and unpinning C:24

qualified names for O:31

refreshing cached results C:32

removing P:38

removing SQL views generated from P:35

restricting row output P:248

rowsets as output of P:274

running C:80

sample XML schema P:280

scheduling for caching C:73

searching for AD:233

database operations (*continued*)

setting JDBC fetch size for P:254

setting permissions P:14

SQL statements in C:86

tagging for on-demand caching P:146

transactional behavior of P:79

unscheduling P:150

uses of P:2

viewing P:13, P:28

viewing dependencies P:32

viewing details about P:29

See also virtual database operations

database operations API API:40

database service SOAP operations

accessibleDBOp API:42

executeAdHocDBOp API:43

executeAdHocDBOpWithOutput API:44

executeAdHocDBOpWithOutputAttach API:46

executeAdHocDBOpWithOutputString API:47

executeDBOp API:48

executeDBOpBytesInput API:49

executeDBOpGridFileInput API:50

executeDBOpWithOutput API:50

executeDBOpWithOutputAttach API:52

executeDBOpWithOutputString API:53

getDBOpOutput API:54

getDBOpOutputAttach API:55

getDBOpParameters API:56

getDBOpSchema API:56

getDBOpSchemaAttach API:57

getDBOpSchemaString API:58

getOutputString API:55

getSQL API:58

listDBConns API:59

listDBOps API:59

listDBOpsByDBConn API:60

database, virtual, See virtual database

DatabaseAdministrators group O:44

DatabaseMetaData interface API:75

databases

Avaki tools for working with O:21

connecting to P:3

for Avaki servers, backing up C:23

protecting O:9

schemas, viewing P:9

supported for connecting to Avaki AD:3, AD:5

DataCatalogAttribute complex type API:14

DataCatalogEntry complex type API:15

DataCatalogPermission complex type API:15

DataProviders group O:44

dataservice --add-schedule command C:48

dataservice --create command C:52

dataservice --delete command C:52

dataservice --delete-schedule command C:54

dataservice --depends command C:54

dataService element P:262

dataservice --execute command C:55

dataservice --generate-sql view command C:56



- dataservice --info command C:57
- dataservice --list-schedules command C:58
- dataservice --update command C:58
- DataServiceExecutionParams complex type API:16
- DB2, versions and JDBC drivers for use with Avaki AD:6
- dbconn --allow-dbop-creation command C:59
- dbconn --delete command C:61
- dbconn --disallow-dbop-creation command C:62
- dbconn --execute command C:63
- dbconn --info command C:64
- dbconn --jdbc command C:66
- dbconn --provision-tables command C:71
- dbconn --show-tables command C:73
- dbconn --test command C:72
- DBOPs, See database operations
- dbop --add-schedule command C:73
- dbop --delete command C:78
- dbop --delete-schedule command C:78
- dbop --depends command C:79
- dbop --execute command C:80
- dbop --generate-sql view command C:82
- dbop --info command C:83
- dbop --jdbc command C:83
- dbop --jdbc --create-virtual-dbop command C:87
- DBOPExecutionParams complex type API:16
- DbopGroupWorkUnit class for data services P:189
- DbopPipeWorkUnit class for data services P:190
- db-path option (DGAS) AD:74
- debug mode, enabling in an Avaki shell C:245
- delimiter character for JDBC schema names API:69
- Deny permission in ACL AD:243, O:47
- dependencies S:22
  - defined AD:353, API:87, C:311, O:65, P:293, S:179
  - listing for data services C:54, P:97
  - listing for database operations C:79, P:32
  - listing for view generators C:272, P:228
  - listing for virtual database operations P:59
- description element P:263
- descriptors for data services P:92
- development tools for web services
  - Apache Axis API:5
  - Microsoft Visual Studio API:5
  - SOAP::Lite API:5
  - VB .NET API:5
- DGAS
  - about AD:62
  - adding user self mappings C:109
  - admission policies AD:332
    - about AD:85
    - adding AD:87, C:91
    - deleting AD:88, C:97
    - displaying C:114
    - displaying Windows domains for C:114
  - admission policies
    - setting defaults C:112
    - setting Windows domains for C:111
    - unsetting Windows domains for C:116
- DGAS (*continued*)
  - changing permissions and ownership AD:334
  - CIFS access to data grid AD:338
  - clearing cached credentials AD:117, C:95
  - configuring associated cache service C:33
  - configuring location of internal caches AD:73
  - configuring to use nondefault ports AD:74, AD:75
  - configuring users and groups AD:67
  - connect port C:224
  - connecting to a domain AD:79, C:223
  - controlling cache size AD:124
  - create CIFS shares C:96
  - default name for AD:73
  - default users, groups, UIDs and GIDs AD:333
  - defined AD:352, API:86, C:310, O:64, P:292, S:178
  - deleting cached objects C:98
  - deleting CIFS shares C:99
  - deleting files and directories from cache AD:119
  - deleting user mappings C:111
  - destroying C:224
  - disconnecting CIFS clients C:101
  - displaying associated cache C:27
  - displaying cache size C:101
  - displaying cache statistics C:102
  - displaying connected CIFS clients C:105
  - displaying free disk space on cache machine C:102
  - displaying information about CIFS shares C:95
  - displaying property descriptions C:104
  - displaying property values C:103
  - dynamic and nondynamic properties AD:83
  - file locking in Unix, interference with AD:64
  - forcing cache to refresh AD:121
  - installing in Unix AD:18
  - installing in Windows AD:24
  - listing CIFS shares C:105
  - listing properties and their values C:103
  - managing cache AD:117
  - mappings
    - default, adding and displaying AD:71
    - for groups, adding AD:70, C:92
    - for groups, deleting C:99
    - for groups, displaying C:106
    - for users, adding AD:70, C:94
    - for users, deleting C:100
    - for users, displaying C:106
    - users and groups, per-DGAS AD:88
    - users, groups, and defaults, domain-wide AD:68
  - modifying CIFS shares C:107
  - NFS clients, not running with AD:64
  - NFS daemons, shutting down before running DGAS AD:66
  - per-DGAS user mappings AD:333
  - ports used by AD:9
  - preparing to start AD:65
  - properties file for AD:76, C:293
  - reading log properties C:107
  - releasing CIFS ports before running AD:66
  - resetting cache statistics C:108

## DGAS (continued)

restarting AD:84  
 saving a copy of the cache C:109  
 saving copies of cache AD:120  
 server logs AD:317  
 setting a cache service AD:90  
 setting location of state database AD:74  
 setting properties AD:82, C:113  
 setting up NFS clients AD:91  
 starting AD:73, C:104  
 stopping C:225  
 syncing cache AD:121  
 time required to upgrade AD:341  
 uncoupling associated cache C:34  
 unsetting properties C:117  
 viewing and resetting cache statistics AD:123  
 warming and updating the cache C:115  
 See also data grid access servers  
 dgas --add-admission-policy command C:91  
 dgas --add-group-mapping command C:92  
 dgas --add-user-mapping command C:94  
 dgas --cifs-share-info command C:95  
 dgas --clear-cached-credentials command C:95  
 dgas command  
   example AD:74  
   syntax AD:73  
 dgas --create-cifs-share command C:96  
 dgas --delete-admission-policy command C:97  
 dgas --delete-cache command C:98  
 dgas --delete-cifs-share command C:99  
 dgas --delete-group-mapping command C:99  
 dgas --delete-user-mapping command C:100  
 dgas --disconnect-cifs-client command C:101  
 dgas --get-cache-size command C:101  
 dgas --get-cache-statistics command C:102  
 dgas --get-free-disk-space command C:102  
 dgas --get-properties command C:103  
 dgas --get-property command C:103  
 dgas --get-property-list command C:104  
 dgas --initialize command C:104  
 dgas --list-cifs-clients command C:105  
 dgas --list-cifs-shares command C:105  
 dgas --list-group-mappings command C:106  
 dgas --list-user-mappings command C:106  
 dgas --modify-cifs-share command C:107  
 dgas --read-log-properties command C:107  
 dgas --register command C:5  
 dgas --reset-cache-statistics command C:108  
 dgas --save-cache command C:109  
 dgas --self-map C:109  
 dgas --self-unmap C:111  
 dgas --set-admission-policy-domain command C:111  
 dgas --set-default-admission-policy command C:112  
 dgas --set-property command C:113  
 dgas --show-admission-policies command C:114  
 dgas --show-admission-policy-domain command C:114  
 dgas --start command C:6

dgas --stop command C:7  
 dgas --sync-cache command C:115  
 dgas --unregister command C:8  
 dgas --unset-admission-policy-domain command C:116  
 dgas --unset-property command C:117  
 directories  
   adding schedule exclusions for caching C:121, C:125  
   adding to cache service C:122  
   Avaki directories, defined AD:349, API:83, C:307, O:61,  
     P:289, S:175  
   Avaki installation API:84, P:290, S:176  
   Avaki installation directory AD:350, C:308, O:62  
   changing C:41  
   changing ownership C:42  
   changing permissions for AD:239  
   copying AD:213, C:46  
   creating AD:208, C:167  
   deleted, purging from cache C:26  
   deleting AD:219, C:195  
   deleting caching schedules C:122  
   displaying name of current directory C:193  
   evicting from cache P:135  
   exporting from the data grid AD:125  
   home, creating AD:169  
   icon for O:29  
   invalidating all in cache C:28  
   invalidating from cache P:136  
   invalidating in cache C:27  
   linking AD:217, C:161  
   listing C:166  
   listing schedules C:126  
   listing those pinned for caching P:134  
   marking for no caching C:126, P:132  
   marking for scheduled caching P:120  
   moving AD:210, C:176  
   NFS-mounting AD:92  
   permissions in O:48  
   purging all from cache and unpinning C:25  
   purging from cache and unpinning C:24  
   refreshing in cache C:32  
   renaming AD:212  
   scheduling for caching C:117  
   searching for AD:233  
   setting ACLs for AD:171  
   shared, See Avaki shares  
   tagging for on-demand caching P:129  
   temp, setting for grid servers AD:135  
   top-level, described O:32  
   unscheduling from cache P:135  
 directory --add-schedule command C:117  
 directory --cache command C:122  
 directory --delete-schedule command C:122  
 directory --do-not-cache command C:126  
 directory --list-schedules command C:126  
 disk space  
   available, displaying for DGAS cache C:102  
   requirements for Avaki software AD:4, AD:16

- distributed transactions
    - about O:25, P:78
    - API for executing P:188
    - configuring database connectors for P:7, S:36
    - defined AD:353, API:87, C:311, O:65, P:293, S:180
    - requirements for P:79
    - setting up P:80
    - supported DBMSes P:79
    - two-phase commit protocol P:79
  - DNS aliases for GDC machines AD:10
  - DNS name, setting for a server AD:32
  - document/literal web services API:3, API:5
  - documentation
    - Avaki, list of AD:xii, API:vi, C:xii, O:vi, P:viii, S:viii
    - for Eclipse Workbench S:12
  - domain --create command C:127
  - domain --disconnect command C:127
  - domain --info command C:128
  - domain --interconnect command C:128
  - domain names, defined AD:354, API:88, C:312, O:66, P:294, S:180
  - Domain Users group in Active Directory C:155
  - domains, Avaki
    - creating C:127
    - defined AD:350, API:84, C:308, O:62, P:290, S:176
    - disconnecting C:127
    - displaying information about AD:98
    - getting information about through JDBC API:75
    - interconnecting AD:289, C:128
    - joining together AD:289
    - naming AD:41, AD:354, API:88, C:312, O:66, P:294, S:180
    - obtaining information about C:128
    - planning before install AD:1
    - providers and consumers AD:289
    - remote, logging in to AD:201
    - specifying for JDBC connections API:69
  - DomainUsers group O:44
  - downstream variables menu S:71
  - downstream, defined S:3
  - drivers, See JDBC drivers
  - drivers directory AD:341
  - dynamic and nondynamic properties (DGAS) AD:83
  - dynamic user mappings
    - creating C:109
    - deleting C:111
- E**
- Eclipse Workbench S:12
  - EII, See enterprise information integration
  - elements
    - connecting S:57
    - descriptions of S:59
    - Input Source S:125
    - moving S:56
  - elements (*continued*)
    - names of S:58
    - operators S:5
    - properties dialogs S:58
    - Result S:164
    - selecting S:56
      - with red borders S:60
    - encryption and HTTPS API:8
    - encryption levels for Avaki shares
      - changing AD:279
      - displaying C:222
      - setting at share creation AD:261
    - encryption of grid objects AD:139
    - enterprise information integration, defined AD:354, API:88, C:312, O:66, P:294, S:180
    - equal signs in command syntax AD:xvi, C:xvi, P:xii
    - error handling S:143
    - errors in view models S:60
    - everyone group O:44, O:48
    - examples
      - conventions used in C:xv
      - data catalog web service API:18
      - data services API API:35
      - database operations API API:42
      - web services clients
        - Java API:9
        - Perl API:9
        - VB .NET API:9
    - exclusions, See schedule exclusions
    - execute inputs in parallel field for Iterator operators S:151
    - Execute JavaScript function for data service plug-ins P:203
    - executeAdHocDBOp SOAP operation API:43
    - executeAdHocDBOpWithOutput SOAP operation API:44
    - executeAdHocDBOpWithOutputAttach SOAP operation API:46
    - executeAdHocDBOpWithOutputString SOAP operation API:47
    - executeDBOp SOAP operation API:48
    - executeDBOpBytesInput SOAP operation API:49
    - executeDBOpGridFileInput SOAP operation API:50
    - executeDBOpWithOutput SOAP operation API:50
    - executeDBOpWithOutputAttach SOAP operation API:52
    - executeDBOpWithOutputString SOAP operation API:53
    - executeDS SOAP operation API:36
    - execution services
      - about AD:109, P:77, S:51
      - configuring AD:109, C:129
      - configuring threads for AD:131
      - defined AD:354, API:88, C:312, O:66, P:294, S:180
      - displaying information about C:129
    - executionservice --info command C:129
    - executionservice --set command C:129
    - executionServiceHint JDBC property API:70
    - exiting from an Avaki session C:165
    - expressions in operators S:4
    - expressions menu, using S:71
    - externalCacheService JDBC property API:62, API:70

## F

- failover
  - defined AD:354, API:88, C:312, O:66, P:294, S:180
  - managing AD:112
  - setting up a secondary GDC AD:43
- fake\_metadata JDBC connection property for ASE API:70
- FAKE\_METADATA JDBC property for ASE AD:7
- federated data access AD:354, API:88, O:66, P:294, S:181
- fetch size, See JDBC fetch size
- file --add-schedule command C:130
- file --cache-on-demand command C:134
- file --delete-schedule command C:135
- file --do-not-cache command C:136
- file --list-schedules command C:136
- file locking AD:64
  - suppressing in NFS mount command AD:93
- file --pin command C:137
- file size attribute C:290
- fileRead SOAP operation API:21
- fileReadAttach SOAP operation API:21
- fileReadString SOAP operation API:22
- files
  - .amm files S:11
  - .avm files S:11
  - .js JavaScript files S:11
  - .jsi JavaScript include files S:11, S:75
  - sample S:115
  - adding schedule exclusions for caching C:133, C:140
  - Avaki\_JDBCStandAlone.jar API:66
  - Avaki\_JDBCStandAlone\_Minus3rd.jar API:66
  - avaki\_studio.properties AD:129
  - AvakiAPI.disco WSDL discovery file API:3
  - AvakiAPIDocLit.wsdl API:3
  - AvakiAPIRpcEnc.wsdl API:3
  - AvakiAPIWithMIMEDocLit.wsdl API:3
  - AvakiAPIWithMIMERpcEnc.wsdl API:3
  - avakijdbc.properties API:67
- bindings.xml
  - copying during upgrade AD:344
  - on grid servers AD:50
  - on proxy servers AD:300
  - on share servers AD:59
- build.xml for data service plug-ins P:184, P:197
- cached, permissions on O:50
- caching of O:56
- changing ownership C:42
- changing permissions for AD:239
- clearing locks C:163
- configuring caching P:107
- configuring encryption level C:216
- connectinfo.txt AD:131
- copying AD:213, C:46
- copying locally AD:215
- data type mapping
  - about C:301
  - command to specify location C:68
  - format of C:301
  - deleted, purging from cache C:26
  - deleting AD:219, C:195
  - deleting pin schedules for C:135
  - DGAS properties AD:76
  - dgas\_log.xml, DGAS log properties file AD:317
  - displaying C:35
  - evicting from cache P:135
  - for data service plug-ins P:180
  - icon for O:29
  - in the data grid O:8
  - install.exe AD:22, AD:24
  - invalidating all in cache C:28
  - invalidating one in cache C:27, P:136
  - JAR files for data service plug-ins P:180
  - jboss-service.xml, request log properties file AD:328
  - jdbc-log4j.properties API:66
  - join.properties file on proxy servers AD:300
  - krb5.conf AD:152
  - linking AD:217, C:161
  - listing C:166
  - listing locks C:164
  - listing schedules C:136
  - listing those pinned for caching P:134
  - log4j.xml,
    - Avaki Studio log properties file AD:317
    - server log properties file AD:317
  - manifest files for data service plug-ins P:180, P:197
  - marking for no caching C:136, P:132
  - marking for scheduled caching P:120
  - moving AD:210, C:176
  - permissions on new files O:49
  - pinning for scheduled caching C:137
  - properties files for DGAS C:293
  - purging all from cache and unpinning C:25
  - purging from cache and unpinning C:24
  - readme AD:12, AD:15, AD:339
  - refreshing in cache C:32
  - renaming AD:212
  - rendering results into O:25
  - scheduling for caching C:130
  - searching for AD:233
  - shareserver.ports
    - on grid servers AD:50
    - on share servers AD:60
  - system.properties AD:129
  - tagging for on-demand caching C:134, P:129
  - temporary, for sorting large result sets S:76
  - unscheduling cached files P:135
  - uploading to the data catalog AD:282
  - Workbench.project S:11
- fileWrite SOAP operation API:23
- Firefox
  - version requirements AD:5
  - setting for selecting run-as users P:27, P:54, P:92, P:224, P:227
- fonts in Studio display, setting S:23

- functions
  - in expressions S:73
  - used with Aggregate operator S:110
- G**
- garbage collection for views C:273
- GDCs, See grid domain controllers
- generated views
  - about O:25, P:217, P:240
  - defined AD:354, API:88, C:312, O:66, P:294, S:181
  - running P:240
  - scheduling updates P:231
  - transactional consistency of P:217
- GeneratedViews directory O:33
- generating schemas S:19
- Generator operator S:117
- getAttributes SOAP operation API:23
- getCatalogs method API:75
- getCatalogTerm method API:75
- getDBOpOutput SOAP operation API:54
- getDBOpOutputAttach SOAP operation API:55
- getDBOpParameters SOAP operation API:56
- getDBOpSchema SOAP operation API:56
- getDBOpSchemaAttach SOAP operation API:57
- getDBOpSchemaString SOAP operation API:58
- getDSOutput SOAP operation API:37
- getDSOutputAttach SOAP operation API:38
- getDSOutputString SOAP operation API:38
- getDSParameters SOAP operation API:39
- getOutputString SOAP operation API:55
- getSchemas method API:75
- getSchemaTerm method API:75
- getSQL SOAP operation API:58
- getSystemAttributes SOAP operation API:24
- getUserAttributes SOAP operation API:24
- GIDs, configuring AD:68, C:217, C:221
- Global Parameters menu S:71
- grid directories, See directories, Avaki
- grid domain controllers
  - backing up and restoring AD:113
  - creating C:127
  - defined AD:355, API:89, C:313, O:67, P:295, S:181
  - DNS aliases for AD:10
  - loading AD:14
  - ports used by AD:6, AD:8
  - primary AD:355, API:89, C:313, O:67, P:295, S:181
  - secondary AD:43
  - starting AD:14
  - stopping AD:38, C:11
  - stopping and restarting GDCs registered as services AD:39
- grid domains
  - See domains, Avaki
- grid servers
  - associating with caches P:111
  - backing up and restoring AD:113
  - choosing for web services API:5
  - configuring AD:48, AD:52
  - grid servers (*continued*)
    - configuring associated cache service C:33
    - configuring cache service threads AD:131
    - configuring nondefault ports AD:50
    - connecting C:225
    - connection info, setting S:23
    - defined AD:355, API:89, C:313, O:67, P:295, S:181
    - destroying C:226
    - disabling auto-restart on C:12
    - disassociating from caches P:112
    - displaying associated cache C:27
    - displaying status of operations on C:248
    - enabling auto-restart C:9
    - finding connect ports AD:53
    - finding server names AD:53
    - installing JDBC drivers on AD:49
    - monitoring AD:99
    - obtaining upgrade information C:250
    - ports used by AD:6, AD:8
    - registering C:9
    - request logs for AD:327
    - server logs AD:317
    - settable attributes of C:290
    - setting location of temp directory for AD:135
    - setting plug-in cache size properties AD:137
    - setting up command clients on AD:94
    - starting AD:50, C:9, C:10
    - stopping AD:51, C:11, C:12, C:227
    - stopping and restarting AD:52
    - time required to upgrade AD:340
    - uncoupling associated cache C:34
    - unregistering C:12
    - upgrading C:249
  - grid user accounts, See users
  - grid-server --register command C:9
  - grid-server --start command C:10
  - grid-server --stop command C:11
  - grid-server --unregister command C:12
  - group --add --user command C:138
  - Group By operator S:76, S:121
  - group --create command C:141
  - group --delete command C:143
  - group --delete --user command C:144
  - group --info command C:145
  - group --list-user command C:147
  - group mappings, adding for a particular DGAS C:92
  - groups
    - about O:43
    - activating privileges for newly added users AD:192, AD:243
    - adding to ACLs AD:243, S:97
    - adding users to AD:191, C:138
    - Administrators AD:45, O:43
    - configuring default mappings C:218
    - creating AD:191, C:141
    - DatabaseAdministrators O:44
    - DataProviders O:44
    - default grid groups O:43
    - default groups for DGAS AD:333

groups (continued)

- defined S:176
- deleting AD:198, C:143
- deleting users from C:144
- displaying information about AD:195, C:145
- enabling interconnection access AD:304
- everyone group O:44
- imported groups O:43
  - from LDAP AD:159
  - from NIS AD:165
  - refreshing AD:195
- in Avaki, defined AD:350, API:84, C:308, O:62, P:290
- listing C:144
- listing users in C:147
- making account changes take effect immediately for DGAS access AD:117
- MessagingUsers O:44
- modifying AD:195
- removing from ACLs AD:242
- removing users from AD:193
- setting up for DGAS AD:67
- UserAdministrators AD:45, O:44
- using in ACLs for cached objects O:50

**H**

hard links

- about O:36
- broken, to generated views P:240
- creating AD:217, C:161
- defined AD:355, API:89, C:313, O:67, P:295, S:182

hardware requirements for Avaki AD:2

help command C:148

help, online, for command line AD:xiii, C:xiii, P:ix

hideCatalogs JDBC property API:70

home directories AD:169

host names

- aliasing for GDCs AD:10
- setting for servers AD:32

HTTP and HTTPS ports

- default AD:8, AD:9, AD:10
- properties for AD:140

HTTP and web services API:7, API:8

HTTP POST problem in web browsers AD:5

HTTP request logs, See request logs

HTTPS and web services API:7, API:8

hyphens in command syntax AD:xvi, C:xvi, P:xii

**I**

IATEMPDIR environment variable AD:16

IBM AIX O:16

IBM AIX requirements AD:3

IBM DB2, versions and JDBC drivers for use with Avaki AD:6

icons for grid objects in the data catalog O:29

id command C:149

identity mapping P:6, S:36

imported groups, See groups

imported user accounts AD:167

increment field for Iterator operators S:151

indent size property for XML files AD:142

initialize field for Iterator operators S:151

initParameter element P:263

inner join S:156

input parameters

- creating P:88
- for data services
  - configuring P:84
  - deleting P:86

Input Source element S:125

input sources

- accessing columns from S:68
- browsing for in data catalog view S:18
- configuring for view models in Avaki Studio S:43
- creating S:55
- error handling S:143
- finding S:18

input streams, for data services, configuring P:87

inputParameter element P:264

inputSource element P:265

InputSource interface for data services P:186

inputStream element P:266

installation directory AD:350, API:84, C:308, O:62, P:290, S:176

installing Avaki

- about AD:13
- in Unix AD:16
- in Windows AD:23
- preparation and planning AD:1
- system requirements AD:2

integration, See data integration

interconnection IDs

- creating AD:295, C:149
- using in permissions AD:304, C:43
- using to provide cross-domain data access AD:295

interconnections between grid domains

- about AD:289
- breaking C:127
- creating AD:291
- defined AD:355, API:89, C:313, O:67, P:295, S:182
- disconnecting domains AD:314
- enabling access AD:299
- prerequisites for AD:94
- setting up C:128
- two-way, exposing users AD:308
- user access methods AD:294
- viewing interconnected domains AD:310

Interconnects directory O:33

Internet Explorer

- Avaki version requirements AD:5
- setting for selecting run-as users P:27, P:54, P:92, P:224, P:227

Intersection operator S:148

- performance S:76

IP address, setting for a server AD:32

isDSAvakiXML SOAP operation API:40

isList element P:266  
 Iterator operator S:149  
 example S:152

## J

### JAR files

for Avaki JDBC driver API:66  
 configuring path for second JAR on one grid server P:247  
 for data service plug-ins P:180  
 for jConnect API:67

jarurl element P:267

### Java

data service plug-in code samples P:190  
 sample web services client API:9  
 writing data service plug-ins in P:183

### Java transformers

error logging P:246  
 implementing P:243  
 installing P:245  
 referring to other documents P:245  
 using P:245

java.io.tmpdir system property AD:135

java.protocol.handler.pkgs system property AD:141

java.rmi.server.hostname system property AD:32

java.security.krb5.conf system property AD:143, AD:153

java.security.krb5.kdc system property AD:153

java.security.krb5.realm system property AD:153

java.sql.DatabaseMetaData interface API:75

Javadoc, Avaki, accessing P:185

### JavaScript

files S:11  
 include files S:11, S:75  
 sample S:115  
 methods on data service plug-in objects P:201  
 resources for learning about S:67  
 using Java classes and interfaces in data service plug-ins P:200  
 writing data service plug-ins in P:200

### JavaScript expressions

about S:4, S:66  
 menu for constructing S:71  
 uses of in Avaki Studio S:vii

jConnect, See JDBC drivers

### JDBC

accessing data catalog through O:24  
 and caching of database operation results API:62  
 connection properties API:68  
 data types API:76  
 directing queries to a particular grid server API:70  
 result set types API:75

### JDBC drivers

about API:61  
 Avaki  
 choosing version of API:66  
 connection properties API:68  
 JAR files for API:66

### JDBC drivers (continued)

#### Avaki

setting classpath for API:67  
 when to use API:65

batch mode API:77

choosing API:65

configuring for a database connector P:5

configuring two versions on one grid server P:247

connection strings API:71

defined AD:356, API:90, C:314, O:68, P:295, S:182

for supported DBMSes AD:3

installing AD:49

#### jConnect

changing default port API:71

setting classpath for API:67

using with Sybase databases AD:6

when to use API:65

loading API:68

prerequisites for using API:64

sample code API:77

setting system properties for API:67

supported features API:74

### JDBC fetch size

setting for database connectors P:5, S:36

setting for database operations P:254

### JDBC schema names API:69

jdbc-log4j.properties file API:66

join algorithms S:155

Join operator S:154

in tutorial S:44

performance S:76

join types S:156

join.properties file on proxy servers AD:300

JRE versions supported by Avaki AD:5

## K

keepalive properties for HTTP ports AD:140

### Kerberos

configuring with LDAP authentication services AD:152

system properties for AD:143

krb5.conf Kerberos configuration file AD:152

## L

last access time file attribute C:290

LBF AD:280

### LDAP

authentication services AD:148

See also authentication services, LDAP

authentication through DGAS AD:86

configuring for Kerberos access AD:152

disabling import on login AD:157

host port, default and nondefault AD:149

importing users on login AD:157

specifying a nondefault host port C:158

ldap --add-schedule command C:149

ldap --delete command C:153  
 ldap --delete-schedule command C:154  
 ldap --import command C:155  
 ldap --info command C:157  
 ldap --integrate command C:157  
 ldap --list-schedules command C:158  
 ldap --searchbase command C:160  
 ldap --update command C:160  
 links  
   command for creating C:161  
   uses of in data catalog O:36  
   See also hard links, soft links  
 Linux requirements AD:3  
 listDBConns SOAP operation API:59  
 listDBOps SOAP operation API:59  
 listDBOpsByDBConn SOAP operation API:60  
 listDomains SOAP operation API:25  
 listDSs SOAP operation API:40  
 listSearches SOAP operation API:25  
 ln command C:161  
 load balancing factor for share servers AD:280, C:244  
 local caches AD:356, API:90, C:314, O:14, O:68, P:296, S:182  
 locks command C:163  
 locks on files  
   clearing C:163  
   obtaining a list C:164  
 log properties file, sample AD:323  
 log4j AD:318, P:196  
 logging  
   audit logs AD:108  
   categories of loggable events AD:318  
   configuring audit logging AD:319  
   for data service plug-ins P:196  
   for JDBC API:66  
   for TrAX transformers P:246  
   for type mapping C:304  
   HTTP request logs AD:108  
   log4j properties files for servers and Studio AD:317  
   properties files for request logs AD:327  
   viewing the server log AD:107  
 logging in AD:199, C:164  
 logical operators S:72  
 logical operators in searches AD:234  
 logicBox element P:268  
 login command C:164  
 login info, setting S:23  
 logout command C:165  
 ls command C:166  
 ls SOAP operation API:26  
 lsSize SOAP operation API:26

## M

manifest files for data service plug-ins P:180, P:197  
 mappings  
   between Avaki and local users/groups for DGAS AD:68  
   between Avaki users and database users, See database identity mappings

mappings (continued)  
 database identity  
   adding C:252  
   deleting C:253  
   listing C:255  
 default, setting up AD:71  
 DGAS default AD:69, C:220  
 DGAS domain-wide  
   for groups, setting up AD:70  
   for users, setting up AD:70  
   users, groups, and defaults AD:68  
 DGAS dynamic C:109  
 for data types in SQL views C:68, C:301  
 for users and groups for DGAS C:94, C:109  
 per-DGAS  
   adding for groups C:92  
   adding for users C:94  
   deleting C:100  
   deleting for groups C:99  
   per-DGAS, users and groups AD:88  
   self mappings for users C:109  
   See also data type mappings  
 maximum concurrent data services setting for execution services AD:109  
 memory requirements for Avaki software AD:3  
 message tests in monitor services AD:101  
 message timeout properties for Avaki servers and clients AD:133  
 MessagingUsers group and user accounts O:44  
 metadata O:13, S:3  
 Metadata directory O:33  
 metadata models  
   about S:77  
   creating S:84  
   defined AD:356, API:90, C:314, O:68, P:296, S:182  
   deleting S:94  
   deploying C:285, S:91  
   deriving S:92  
   editing S:84  
   files associated with S:11  
   importing S:79  
   mapping to Avaki objects S:88  
   naming scheme S:91  
   undeploying C:286, S:94  
 Microsoft SQL Server, versions and JDBC drivers for use with Avaki AD:6  
 Microsoft Visual Studio API:5  
 MicroSoft Windows O:16  
 MIME in Avaki web services API:3  
 minus signs in command syntax AD:xvi, C:xvi, P:xii  
 mkdir command C:167  
 mkdir SOAP operation API:27  
 mkdirParents SOAP operation API:27  
 mkdirServer SOAP operation API:29  
 mkdorParentsServer SOAP operation API:28  
 models, See metadata models and view models  
 modification time file attribute C:290



monitor --add command C:167  
 monitor --clear command C:168  
 monitor --create command C:170  
 monitor --delete command C:171  
 monitor --list command C:172  
 monitor --result command C:172  
 monitor services  
 monitor --start command C:173  
 monitor --stop command C:174  
 monitoring  
   about AD:99  
   adding tests C:167  
   configuring AD:101  
   creating monitor services C:170  
   deleting monitor services AD:106, C:171  
   deleting tests AD:105  
   disabling and enabling tests AD:104  
   listing active tests C:172  
   logging AD:107  
   message tests AD:101  
   ping tests AD:100  
   removing tests C:168  
   restarting tests AD:105, C:173  
   stopping tests AD:105, C:174  
   viewing results AD:103, C:172  
 mount port for DGAS AD:81  
 mount protocol port, default AD:9  
 Mozilla, Avaki version requirements AD:5  
 Multiplexer operator S:157  
 multiplexing socket properties AD:142  
 mv command C:176  
 mv SOAP operation API:29  
 MySQL  
   configuring XA driver for P:7, S:36  
   versions and JDBC drivers for use with Avaki AD:6

## N

name element P:269  
 name property for columns S:61  
 names of Avaki objects  
   about O:29  
   avoiding underscores in when using BusinessObjects API:81  
   case sensitivity and restrictions AD:206  
   changing AD:212  
   metadata models and mapped tables S:91  
   of elements S:58  
   qualified names O:30  
   restrictions on AD:41  
   three-part O:24  
 navigator in Studio S:17  
 nesting operations in data services S:149  
 .NET, See .NET under Symbols at the beginning of the index  
 Netscape requirements AD:5  
 NFS  
   and permissions AD:12  
   configuring NFS port for DGAS AD:81

NFS (*continued*)  
   port, default AD:9  
   shutting down before starting a DGAS AD:65  
 NFS clients  
   attribute caching for AD:336  
   defined AD:356, API:90, C:314, O:68, P:296, S:183  
   older, accessing data grid through AD:337  
   requirements for O:16  
   setting up AD:91  
 NFS URLs AD:93  
 NIS  
   disabling import on login AD:163  
   importing users on login AD:163  
   See also authentication services, NIS  
 nis --add-schedule command C:177  
 nis --delete command C:181  
 nis --delete-schedule command C:181  
 nis --import command C:182  
 nis --info command C:183  
 nis --integrate command C:184  
 nis --list-schedules command C:184  
 nis --update command C:185  
 NLM AD:64  
 NLM protocol port, default AD:9  
 notifications, See update notifications

## O

object host name grid server attribute C:290  
 octothorpe AD:33, AD:129  
 ODBC  
   accessing data catalog through O:24  
   defined AD:356, API:90, C:314, O:68, P:296, S:183  
   support for API:80  
 ODBC drivers, using with Avaki API:80  
 offline expiration cache attribute C:291  
 on-demand caching  
   about P:119  
   defined AD:357, API:91, C:315, O:69, P:296, S:183  
   of database operation and data service results O:57, P:108  
   of files O:56, P:107  
 online help for command line AD:xiii, C:xiii, P:ix  
 operating systems supported by Avaki O:16  
 operations, monitoring AD:99  
 operators  
   about S:5  
   adding to a view model S:55  
   Aggregate S:108  
   connecting S:57  
   Custom S:111  
   descriptions of S:59  
   Generator S:117  
   Group By S:121  
   in searches AD:234  
   Input Source S:125  
   Intersection S:148  
   Iterator S:149  
   Join S:154

- operators (*continued*)
    - logical, in expressions S:72
    - moving S:56
    - Multiplexer S:157
    - names of S:58
    - Order By S:159
    - performance considerations S:5
    - Projection S:161
    - properties dialogs S:58
    - Result S:164
    - Select S:165
    - selecting S:56
    - sort-based, performance of S:76
    - Splitter S:166
    - Union S:168
    - Update S:169
    - with red borders S:60
  - Oracle 10g
    - configuring XA driver for P:7, S:36
    - versions and JDBC drivers for use with Avaki AD:6
  - Order By operator S:159
    - performance S:76
  - os arch grid server attribute C:290
  - os name grid server attribute C:290
  - outer-full join S:156
  - outer-left join S:156
  - outer-right join S:156
  - output streams, for data services, configuring P:86
  - outputStream element P:269
  - ownership of objects in the data catalog
    - about O:46
    - setting AD:242
- P**
- palette in Avaki Studio view model editor S:20, S:54
  - parameters
    - accessing in expressions S:71
    - adding S:64
    - deleting S:65
    - displaying S:63
    - for data service plug-ins
      - about P:178
      - specifying for Java plug-ins C:188
      - specifying for XSLT plug-ins P:181
    - for testing view models S:50
    - in Avaki Studio, about S:4
    - input, for data services, configuring P:84
    - mapping input parameters S:144
    - modifying S:65
    - reordering S:65
    - validating S:64
  - ParameterSpec interface for data services P:187
  - passwords
    - changing AD:175, C:185
    - specifying for JDBC connections API:69
  - patches, settable attributes of C:290
  - performance S:5, S:76
    - benefits of caching O:54
    - tracking, enabling in an Avaki shell C:245
  - Perl, sample web services client API:9
  - permissions
    - about O:45
    - changing AD:239, C:41
    - granted by grid groups O:43
    - hiding objects with O:49
    - on new Avaki shares AD:261
    - on shared data AD:12
    - setting in ACLs S:97
    - values for AD:242, O:47
    - viewing for Avaki services C:186
    - See also ACLs
  - permissions command C:186
  - permissions SOAP operation API:30
  - perspectives in Avaki Studio
    - Avaki S:13
    - defined S:12
    - Resource S:11
  - pin for caching, defined AD:357, API:91, C:315, O:69, P:297, S:183
  - ping tests in monitor services AD:100
  - planning an Avaki deployment AD:1
  - platforms supported by Avaki O:16
  - plugin command P:184
  - plugin --generate command C:187
  - Plugin interface for data services P:186
  - Plug-in Wizard and creating data service plug-ins P:183
  - plug-ins, See data service plug-ins
  - plus signs in command syntax AD:xvi, C:xv, P:xi
  - ports
    - bad port cache AD:141
    - changing, See ports, nondefault
    - CIFS, releasing before running a DGAS AD:66
    - default, for Avaki servers AD:6
    - HTTP and HTTPS, See HTTP and HTTPS ports
    - LDAP host
      - default and nondefault AD:149
      - specifying C:158
    - NFS, default AD:9
    - nondefault
      - configuring for DGAS AD:74, AD:75
      - configuring for grid servers AD:50
      - configuring for proxy servers AD:300
      - configuring for share servers AD:59
    - RMI, See RMI ports
    - SMB, default AD:9
    - specifying in WSDL API:6
    - SSL, See SSL ports
    - TDS AD:8, AD:50, AD:145, API:71
  - ports, connect, See connect ports
  - POST problem in web browsers AD:5
  - precision property for columns S:61
  - preferences for Avaki Studio, setting S:23
  - primary GDCs AD:357, C:315, O:69

- privacy and HTTPS API:8
- Projection operator S:161
  - in tutorial S:46
- projects
  - creating S:13
  - defined S:12
- properties
  - cache sizes for data service plug-ins AD:137
  - DGAS
    - configuring AD:82
    - controlling cache size AD:124
    - displaying AD:124, C:103
    - displaying descriptions C:104
    - listing C:103
    - properties file C:293
    - setting C:113
    - unsetting C:117
  - for cache services AD:135
  - for encryption of grid objects AD:139
  - for HTTP and HTTPS ports AD:140
  - for HTTP keepalives on proxy servers AD:140
  - for JDBC clients AD:128
  - for JDBC connections API:68
  - for Kerberos AD:143
  - for multiplexing sockets AD:142
  - for remote object stub cache AD:144
  - for schedule exclusion cache AD:144
  - for server request logs AD:327
  - for servers acting as clients AD:128
  - for share servers AD:138
  - for virtual database table cache size AD:144
  - for XA connections C:70, P:7, S:37
  - Java system properties, providing to JDBC driver API:67
  - JDBC, specifying for a database connector C:67
  - Kerberos default realm AD:153
  - Kerberos key distribution center AD:153
  - location of Kerberos configuration file AD:153
  - message timeouts for Avaki servers and clients AD:133
  - remote coherence window for configurations AD:141
  - setting server's host name or IP address AD:33
  - system. See system properties.
  - views
    - displaying C:274
    - setting C:275
  - XML indent size AD:142
- properties dialog boxes S:58
- provisioning data O:21
- proxy --add command C:191
- proxy --delete command C:191
- proxy --list command C:192
- proxy routing tables
  - about AD:289
  - configuring AD:292, C:191
  - displaying C:192

- proxy servers
  - about AD:289
  - configuring AD:299
  - configuring nondefault ports AD:300
  - connecting C:228
  - defined AD:357, API:91, C:315, O:69, P:297, S:183
  - deleting from the routing table C:191
  - destroying C:228
  - disabling auto-restart C:14
  - enabling auto-restart AD:300, C:12
  - finding connect port numbers AD:303
  - finding server names AD:303
  - installing in Windows AD:24
  - installing in Unix AD:18
  - ports used by AD:10
  - registering for auto-restart AD:302, C:12
  - request logs for AD:327
  - server logs for AD:317
  - setting HTTP keepalive properties for AD:140
  - setting up C:191
  - starting AD:301, C:12, C:13
  - stopping C:14, C:228
  - stopping and restarting AD:301
  - time required to upgrade AD:341
  - unregistering C:14
  - when to deploy AD:290
- proxy-server --register command AD:302, C:12
- proxy-server --start command C:13
- proxy-server --stop command C:14
- proxy-server --unregister command C:14
- pwd command C:193

## Q

- qualified names
  - about O:30
  - for data services, specifying API:35, C:51, C:267
  - for database connectors, specifying C:254
  - for database operations, specifying C:25, C:264
  - for groups, specifying C:43, C:142, C:145, C:147, C:148
  - for users, specifying C:42, C:53, C:60, C:86, C:89, C:141, C:145, C:165, C:251, C:266, C:280
- queries, See ad-hoc queries
- query engine
  - defined AD:357, API:91, C:315, O:69, P:297, S:184
  - mapping data types for C:301
  - sort chunk size property AD:139
- queryCacheTTL JDBC property API:71

## R

- RAM requirements for Avaki software AD:3
- range input for Iterator operators S:151
- readme file AD:12, AD:15, AD:339
- red borders on operators in Studio S:60
- Red Hat Linux requirements AD:3
- ref element P:270

refresh schedules  
 for Avaki shares AD:266, C:231  
   advanced AD:272  
   calendared AD:271  
   exclusions for AD:274  
   listing C:239  
   one-time AD:269  
   periodic AD:270  
   removing AD:278, C:236  
 for data services P:152  
   advanced P:157  
   calendared P:155  
   one-time P:155  
   periodic P:157  
 for database operations P:139  
   advanced P:144  
   calendared P:142  
   one-time P:142  
   periodic P:144  
 for files or directories P:120  
   advanced P:127  
   calendared P:125  
   one-time P:125  
   periodic P:127  
 for generated views P:231  
   advanced P:236  
   calendared P:234  
   one-time P:233  
   periodic P:236  
 for LDAP authentication services AD:185  
 for virtual database operations P:139  
   advanced P:144  
   calendared P:142  
   one-time P:142  
   periodic P:144  
 refreshing users on login AD:149  
 reindex interval for search services AD:232  
 remote caches AD:357, API:91, C:315, O:14, O:69, P:297, S:184  
 removeAttribute SOAP operation API:31  
 replica --add command C:193  
 replica --config command C:193  
 replica --delete command C:194  
 replica --info command C:194  
 replica --synch command C:195  
 request logs  
   configuring AD:327  
   viewing AD:108  
 requirements, pre-installation AD:2  
 Result element S:164  
 result sets  
   accessor functions S:73  
   combining S:44  
   large, providing space for sorting S:76  
   types in JDBC API:75

rm command C:195  
 rm SOAP operation API:31  
 RMI ports  
   default AD:8, AD:9, AD:10  
   linked to grid server connect ports AD:50  
   linked to share server connect ports AD:60  
 routing tables, configuring AD:292, C:191  
 row-level access control S:74  
 RowSetFactory class for data services P:188  
 rowsets O:11, P:273  
 rpc/encoded web services API:3, API:5  
 rpcinfo command AD:66  
 Rudi port AD:352, C:310, O:64  
 run-as users, See users and user accounts

## S

Saxon C:265, C:268, C:271, P:220, P:224, P:227  
 scale property for columns S:61  
 schedule --delete command C:196  
 schedule exclusions  
   about AD:274, P:166  
   adding for Avaki share rehashes C:234  
   adding for data services C:51  
   adding for directories C:125  
   adding for files C:133, C:140  
   adding for LDAP authentication services C:152  
   adding for NIS authentication services C:180  
   adding for views C:262  
   adding to Avaki directories C:121  
   adding to database operations C:76  
   applying to schedule entries P:168  
   caching properties for AD:144  
   configuring AD:274, P:166  
   creating custom C:198  
   creating daily C:199  
   creating monthly C:201  
   creating weekly C:203  
   creating yearly C:205  
   defined AD:358, API:92, C:316, O:70, P:298, S:184  
   deleting C:207  
   displaying information about C:208, P:171  
   listing names C:209  
 schedule --info command C:197  
 schedule --list command C:197  
 schedule --print-iterations command C:198  
 scheduled caching  
   about P:119  
   defined AD:357, API:91, C:315, O:69, P:297, S:184  
   of database operation and data service results O:59, P:109  
   of files O:57, P:107  
 scheduleexclusion --create --custom command C:198  
 scheduleexclusion --create --daily command C:199  
 scheduleexclusion --create --monthly command C:201  
 scheduleexclusion --create --weekly command C:203  
 scheduleexclusion --create --yearly command C:205  
 scheduleexclusion --delete command C:207  
 scheduleexclusion --info command C:208

- scheduleexclusion --list command C:209
- schedules
  - adding for data services C:48, P:152
  - adding for database operations C:73, P:139
  - adding for directories C:117
  - adding for files C:130
  - adding for LDAP user importation C:149
  - adding for NIS user importation C:177
  - adding for views C:259
  - adding for virtual database operations P:139
  - creating cron specifications C:297
  - creating custom exclusions C:198
  - creating daily exclusions C:199
  - creating monthly exclusions C:201
  - creating weekly exclusions C:203
  - creating yearly exclusions C:205
  - cron expressions in AD:185, AD:273
  - deleting C:196
  - deleting exclusions C:207
  - deleting for data services C:54
  - deleting for database operations C:78
  - deleting for directories C:122
  - deleting for files C:135
  - deleting for LDAP user importation C:154
  - deleting for NIS user importation C:181
  - deleting for views C:272
  - displaying exclusion information C:208
  - displaying information about C:197
  - exclusions, see also schedule exclusions
  - execute permissions required O:59
  - for refreshing LDAP authentication services AD:185
  - for view generators and generated views P:231
  - listing C:197
  - listing execution times C:198
  - listing for data services C:58
  - listing for database operations C:88
  - listing for directories C:126
  - listing for files C:136
  - listing for LDAP authentication services C:158
  - listing for NIS authentication services C:184
  - listing for views C:274
  - listing names of exclusions C:209
  - refresh, See refresh schedules
  - types of AD:185, AD:267
- schemas
  - about S:3, S:60
  - enabling browsing on a database connector P:4
  - expressed in metadata models S:77
  - for Avaki data services API:75
  - for operators, column properties of S:60
  - for virtual database operations and their SQL views API:75
  - for virtual database operations, generating P:57
  - generating S:19
  - generating for data services C:55, P:98
  - generating for database operations C:80, P:31
  - getting information about through JDBC API:75
  - getting via JDBC API:75
- schemas (*continued*)
  - JDBC schema names API:69
  - modifying S:161
  - types in Avaki domains P:51, P:64
  - viewing for databases P:9
  - See also metadata models
  - search (execute) command C:211
  - search --create command C:209
  - search --delete command C:210
  - search --get-rehash-level command C:212
  - search --info command C:214
  - search --rehash command C:215
  - search --set-rehash-level command C:215
  - search SOAP operation API:32
  - searches O:13
    - configuring rehash intervals C:215
    - creating search services AD:231, C:209
    - deleting AD:236
    - deleting search services C:210
    - displaying search service information C:214
    - performing AD:233, C:211
    - rehashing search services C:215
    - reindex interval, setting AD:232
    - viewing rehash intervals C:212
  - SearchQuery complex type API:17
  - SearchResult complex type API:17
  - secondary GDCs AD:43
    - adding C:193
    - deleting C:194
    - forcing updates C:195
    - setting refresh intervals C:193
    - setting update interval C:195
    - viewing C:194
    - See also grid domain controllers
  - security
    - .NET API:9
    - about O:10
    - ACLs O:45
    - authentication O:41
    - configuring encryption levels C:216
    - displaying encryption levels C:222
    - for web clients API:8
    - HTTPS API:8
    - setting permissions C:41
    - SSL certificates API:8
    - user accounts and groups O:43
    - viewing permissions C:186
  - security --config command C:216
  - security --default-gid command C:217
  - security --default-group command C:218
  - security --default-uid command C:219
  - security --default-user command C:220
  - security --gid command C:221
  - security --info command C:222
  - security --uid command C:222
  - Select operator S:165
  - server connect ports, See connect ports
  - server --dgas --connect command C:223

- server --dgas --destroy command C:224
- server --dgas --stop command C:225
- server --grid --connect command C:225
- server --grid --destroy command C:226
- server --grid --stop command C:227
- server logs
  - configuring AD:317
  - viewing AD:107
- server --proxy command C:228
- server --share --connect command C:228
- server --share --disconnect command C:229
- server --share --stop command C:230
- servers, backing up databases for C:23
- servers, Avaki
  - defined AD:350, API:84, C:308, O:62, P:290, S:176
  - displaying software version of AD:99
  - finding names of AD:131
  - in a typical deployment O:18
  - monitoring AD:99
  - ports used by AD:6
  - problems communicating with AD:32
  - setting cache service properties for AD:135
  - setting host name or IP address to advertise AD:32
  - setting message timeout properties for AD:133
  - See also DGAS, grid domain controllers, grid servers, share servers, and proxy servers
- servers, proxy, See proxy servers
- services, Avaki
  - copying C:46
  - defined AD:358, API:92, C:316, O:70, P:298, S:184
  - icon for O:29
- setAttribute SOAP operation API:32
- setOutputStream JavaScript method for data service plugins P:202
- share --add-rehash-schedule command C:231
- share --add-share-servers command C:232
- share --create command C:235
- share --delete-rehash-schedule command C:236
- share --disconnect command C:238
- share --get-local-path command C:238
- share --get-status command C:239
- share --list-rehash-schedules command C:239
- share --list-share-servers command C:240
- share --rehash command C:240
- share --remove-share-servers command C:241
- share servers
  - about AD:54
  - adding to Avaki shares AD:263
  - backing up and restoring AD:115
  - before disconnecting AD:54, C:230
  - configuring a machine with one share server AD:55
  - configuring multiple share servers on one machine AD:59
  - configuring to use nondefault ports AD:59
  - connecting to grid servers C:228
  - defined AD:358, API:92, C:316, O:70, P:298, S:184
  - disabling auto-restart C:17
  - disconnecting from grid servers C:229
  - enabling auto-restart C:15
  - finding connect ports AD:59
  - finding server names AD:59
  - installing in Windows AD:24
  - installing on Unix AD:18
  - local path AD:261
  - modifying load balance factor C:244
  - multiple AD:55, AD:263
  - obtaining upgrade information C:250
  - ports used by AD:9
  - registering C:15
  - registering for auto-restart AD:57
  - removing from Avaki shares C:241
  - removing from shares AD:265
  - replacing for Avaki shares C:242
  - server logs for AD:317
  - setting for Avaki shares AD:260
  - setting load balancing factor AD:280
  - setting system properties for AD:138
  - starting AD:56, C:15, C:16
  - stopping C:17, C:230
  - stopping and restarting AD:57
  - time required to upgrade AD:341
  - unregistering C:17
  - upgrading C:249
  - write access and user accounts AD:12
- share --set-local-path command C:241
- share --set-share-servers command C:242
- share --set-status command C:243
- share --update-share-servers command C:244
- shared directories, See Avaki shares
- shares, See Avaki shares and CIFS shares
- Shares directory O:34
- shares, CIFS AD:125
- share-server --register command AD:57, C:15
- share-server --start command C:16
- share-server --stop command C:17
- share-server --unregister command C:17
- shareserver.ports file
  - on grid servers AD:50
  - on share servers AD:60
- shell command C:245
- shells, Avaki, accessing C:245
- shortcuts created in Windows installations AD:27
- SMB ports, default AD:9
- SOAP
  - formal definition API:1
  - learning about API:1
  - over HTTP API:8
  - over HTTPS API:8
  - standards compliance API:1
- SOAP clients, See web services clients
- SOAP complex types API:12
  - AdHocDBOPExecutionParams API:13
  - AvakiPrincipal API:13
  - DataCatalogAttribute API:14
- share servers (*continued*)
  - enabling auto-restart C:15
  - finding connect ports AD:59
  - finding server names AD:59
  - installing in Windows AD:24
  - installing on Unix AD:18
  - local path AD:261
  - modifying load balance factor C:244
  - multiple AD:55, AD:263
  - obtaining upgrade information C:250
  - ports used by AD:9
  - registering C:15
  - registering for auto-restart AD:57
  - removing from Avaki shares C:241
  - removing from shares AD:265
  - replacing for Avaki shares C:242
  - server logs for AD:317
  - setting for Avaki shares AD:260
  - setting load balancing factor AD:280
  - setting system properties for AD:138
  - starting AD:56, C:15, C:16
  - stopping C:17, C:230
  - stopping and restarting AD:57
  - time required to upgrade AD:341
  - unregistering C:17
  - upgrading C:249
  - write access and user accounts AD:12

SOAP complex types (*continued*)

- DataCatalogEntry API:15
- DataCatalogPermission API:15
- DataServiceExecutionParams API:16
- DBOPExecutionParams API:16
- SearchQuery API:17
- SearchResult API:17

SOAP operations

- accessibleDBOp API:42
- accessibleDS API:36
- accessiblePath API:19
- chmod API:19
- chown API:20
- components of web service API:4
- data catalog API:18
- data services API:34
- database operations API:40
- executeAdHocDBOp API:43
- executeAdHocDBOpWithoutOutput API:44
- executeAdHocDBOpWithoutOutputAttach API:46
- executeAdHocDBOpWithoutOutputString API:47
- executeDBOp API:48
- executeDBOpBytesInput API:49
- executeDBOpGridFileInput API:50
- executeDBOpWithOutput API:50
- executeDBOpWithOutputAttach API:52
- executeDBOpWithOutputString API:53
- executeDS API:36
- fileRead API:21
- fileReadAttach API:21
- fileReadString API:22
- fileWrite API:23
- getAttributes API:23
- getDBOpOutput API:54
- getDBOpOutputAttach API:55
- getDBOpParameters API:56
- getDBOpSchema API:56
- getDBOpSchemaAttach API:57
- getDBOpSchemaString API:58
- getDSOutput API:37
- getDSOutputAttach API:38
- getDSOutputString API:38
- getDSParameters API:39
- getOutputString API:55
- getSQL API:58
- getSystemAttributes API:24
- getUserAttributes API:24
- isDSAvakiXML API:40
- listDBConns API:59
- listDBOps API:59
- listDBOpsByDBConn API:60
- listDomains API:25
- listDSs API:40
- listSearches API:25
- ls API:26
- lsSize API:26
- mkdir API:27
- mkdirParents API:27

SOAP operations (*continued*)

- mkdirParentsServer API:28
- mkdirServer API:29
- mv API:29
- permissions API:30
- removeAttribute API:31
- rm API:31
- search API:32
- setAttribute API:32
- tester API:33
- whoami API:33

SOAP::Lite API:5

sockets, multiplexing AD:142

- setting maximum channels AD:142
- setting maximum write AD:142
- setting send buffer size for AD:143

soft links

- about O:36
- creating AD:217, C:161
- defined AD:358, API:92, C:316, O:70, P:298, S:185
- not used in searches AD:233

software requirements for Avaki AD:2

Solaris requirements AD:3

sort chunk size

- controlling S:76
- for query engine AD:139

spaces

- in Windows install pathnames, avoiding AD:26
- to separate arguments in Avaki commands AD:xvi, C:xvi, P:xii

SPARC/Solaris requirements AD:3

Splitter operator S:166

SQL

- aggregate functions and aliasing columns S:42
- as prerequisite for Avaki Studio users S:vii
- statements
  - in database operations C:86, P:251
  - in virtual database operations C:90

SQL Server, versions and JDBC drivers for use with Avaki AD:6

sql view --delete command C:246

sql view --get-description command C:246

sql view --set-description command C:247

SQL views

- about O:8, O:22, P:38
- adding to categories P:47
- configuring attributes P:44
- data type mappings for C:68, C:301, P:39
- defined AD:358, API:92, C:316, O:70, P:298, S:185
- deleting C:246
- displaying descriptions C:246
- displaying tables provisioned from database connectors C:73
- enabling provisioning on a database connector P:4
- from data service results, generating P:60, P:100
- generated from database operations, removing P:35
- generating from data services C:56
- generating from database operations C:82, P:34
- location in categories S:18

- SQL views (*continued*)
    - managing P:20
    - modifying P:43
    - modifying descriptions C:247
    - names in data catalog O:24
    - provisioning P:39
    - provisioning from database connectors C:71
    - qualified names for O:31
    - removing P:44
    - schema types for P:51, P:64
    - schemas for S:22
    - searching for AD:233
    - table types for API:75
    - viewing P:42
    - viewing and modifying ACLs P:46
  - square brackets in command syntax AD:xv, C:xv, P:xi
  - SSL certificates
    - about API:8
    - generating AD:33
    - installing AD:39
    - planning for AD:8
  - SSL ports, default AD:8, AD:9, AD:10
  - status command C:248
  - status of grid operations, monitoring AD:99
  - stored procedures API:73
  - StreamingRowSet interface for data services P:187
  - streams, closing in data service plug-ins P:186
  - Studio, See Avaki Studio
  - style sheet engines
    - for data service view generators P:227
    - for database operation view generators P:224
    - for file view generators P:220
  - style sheet engines for database view generators C:265, C:268
  - style sheet engines for file view generators C:271
  - Sun JDK for compiling data service plug-ins P:184
  - Sun ONE Directory Server AD:148
    - See also authentication services, LDAP AD:148
  - Sun Solaris requirements AD:3
  - SuSE Linux requirements AD:3
  - Sybase ASA, versions and JDBC drivers for use with Avaki AD:6
  - Sybase ASE
    - configuring XA driver for P:7, S:36
    - connection property required for API:70
    - versions and JDBC drivers for use with Avaki AD:7
  - Sybase contact information AD:xvi, API:vii, C:xvi, O:vi, P:xii, S:ix
  - Sybase IQ, versions and JDBC drivers for use with Avaki AD:7
  - syntax conventions for commands AD:xv, C:xiv, P:x
  - system attributes AD:248, S:101
  - System directory O:34
  - system properties
    - about AD:128
    - descriptions of AD:131
    - setting on Avaki Studio AD:129
    - setting on clients AD:129
    - setting on servers AD:129
    - setting with JDBC driver API:67
  - system requirements for Avaki software AD:2, O:16
  - system.properties file AD:33, AD:129
- ## T
- table schema view S:22
  - table types for SQL views API:75
  - tables
    - deleting SQL views C:246
    - displaying descriptions C:246
    - generating from data services C:56
    - generating from database operations C:82
    - in metadata models S:77
      - arranging in editor S:84
      - making accessible via JDBC S:91
      - mapping to Avaki objects S:88
      - naming scheme for S:91
    - in virtual database, displaying C:283
    - mapping data types for C:301
    - modifying descriptions C:247
    - provisioned from database connectors, displaying C:73
    - provisioning as SQL views O:22
    - provisioning from database connectors C:71
    - qualified names for O:32
    - schemas for, displaying S:22
    - See also SQL views
  - tabs for view models S:21
  - target element P:270
  - TCP channel sockets AD:142
  - TDS port
    - changing AD:50, AD:145, API:71
    - default AD:8
  - technical support contact information AD:xvi, API:vii, C:xvi, O:vi, P:xii, S:ix
  - temp directories for grid servers, setting AD:135
  - Templates class P:244
  - tester SOAP operation API:33
  - testing an upgraded grid domain AD:346
  - testing function for WS clients API:33
  - text conventions C:xiv
  - timeout properties for Avaki server communications AD:133
  - timeouts, configuring for database operations P:253
  - transactions, See distributed transactions P:78
  - TrAX API P:243
  - two-phase commit protocol P:79
  - type element P:270
  - type mapping, See data types, mapping
  - type property for columns S:61
  - TypeMapping log4j category C:304
  - types for variables S:70
- ## U
- UID attribute in LDAP authentication services AD:150
  - UIDs, configuring AD:68, C:219, C:222
  - underscore characters in Avaki names API:81
  - Union operator S:168
  - Unix file mode semantics, setting AD:337



Unset permission in ACL AD:242, O:47  
 update intervals for GDCs, setting C:195  
 update notifications  
   configuring P:238  
   defined AD:359, API:93, C:317, O:71, P:299, S:185  
   enabling AD:311  
 Update operator S:169  
 upgrade command C:249  
 upgrade --info command C:250  
 upgrading Avaki software  
   interoperability of different versions AD:339  
   preparation steps AD:341  
   upgrade planning AD:340  
   upgrade steps AD:342  
 upstream, defined S:3  
 urlLogicBox element P:271  
 user attributes AD:248, S:101  
 user command C:250  
 user --create command C:251  
 user --db-mapping --add command C:252  
 user --db-mapping --delete command C:253  
 user --db-mapping --list command C:255  
 user --delete command C:257  
 user groups, See groups  
 user --info command C:258  
 user --list-group command C:258  
 UserAdministrators group AD:45, O:44  
 users and user accounts  
   about O:43  
   adding database identity mappings C:252  
   adding to ACLs AD:243, S:97  
   adding to groups AD:191, C:138  
   administrative accounts, setting up AD:44  
   changing passwords AD:175, C:185  
   clearing credentials from DGAS cache AD:117  
   configuring associated cache service C:33  
   configuring default mappings C:220  
   configuring dynamic mappings C:109  
   configuring self mappings C:109  
   creating accounts AD:168, C:251  
   creating home directories AD:169  
   default users for DGAS AD:333  
   deleting accounts AD:189, C:257  
   deleting database identity mappings C:253  
   deleting from groups C:144  
   disabling import on login (LDAP) AD:157  
   disabling import on login (NIS) AD:163  
   displaying and changing account information AD:187  
   displaying associated cache C:27  
   displaying full names and contact information C:258  
   displaying names C:286  
   enabling interconnection access AD:304  
   enabling on authentication services C:250  
   exposing in a two-way interconnect AD:308  
   giving access to other domains AD:294  
   grid accounts AD:167  
   imported accounts AD:167

users and user accounts (*continued*)  
   importing from LDAP services AD:157  
   importing from NIS services AD:164  
   importing on login (LDAP) AD:157  
   importing on login (NIS) AD:163  
   listing database identity mappings C:255  
   listing group membership for C:258  
   logging in AD:199  
   logging out when newly added to groups AD:192, AD:243  
   making account changes take effect immediately for DGAS  
     access AD:117  
   managing AD:167  
   mapping Avaki users to database users, See database identity  
     mappings  
   MessagingUser O:44  
   qualified names for O:32  
   querying whether enabled in LDAP AD:151  
   refreshing imported accounts AD:185  
   refreshing on login (LDAP) AD:149  
   removing from ACLs AD:242  
   removing from groups AD:193  
   roles for O:43  
   run-as users  
     browser setting for selecting P:27, P:54, P:92, P:224, P:227  
     for data service views P:227  
     for data services P:92  
     for database operation views P:223  
     for database operations P:26  
     for virtual database operations P:54  
   setting run-as user for views C:279  
   setting up for DGAS AD:67  
   setting up local accounts for Avaki AD:11  
   specifying for JDBC connections API:69  
   uncoupling associated cache C:34

## V

validation error expressions S:65  
 validation expressions S:64  
 value element P:272  
 values element P:272  
 variables  
   about S:69  
   allowed types for S:70  
   downstream, menu of S:71  
   in Avaki Studio, about S:4  
   updating S:69  
 VB .NET, See .NET  
 versions  
   of Avaki software, displaying AD:99, C:148, C:250  
   SOAP API:1  
   TrAX P:243  
   WSDL API:1  
 vertical bars in command syntax AD:xv, C:xv, P:xi  
 view --add-schedule command C:259  
 view --create --database command C:263  
 view --create --data-service command C:266

- view --create --file command C:267
- view --delete command C:272
- view --delete-schedule command C:272
- view --depends command C:272
- view --garbage-collect command C:273
- view generators
  - about O:8, O:25, P:217
  - caching of input files P:241
  - configuring update notifications for P:238
  - defined AD:359, API:93, C:317, O:71, P:299, S:185
  - for data services
    - setting up P:225
    - specifying a style sheet engine P:227
  - for database operations
    - setting up P:221
    - specifying a style sheet engine P:224
  - for files
    - setting up P:218
    - specifying a style sheet engine P:220
  - for large data sets and unsupported formats P:242
  - listing dependent operations P:228
  - modifying P:229
  - non-XSLT-based P:242
  - removing P:239
  - rowsets as inputs of P:275
  - running P:240
  - scheduling updates P:231
  - troubleshooting P:240
  - using TrAX transformers P:242
- view --info command C:274
- view --list-schedules command C:274
- view models
  - about O:23, S:2
  - configuring input sources S:43
  - creating S:42
  - defined AD:359, API:93, C:317, O:71, P:299, S:185
  - deploying as data services S:50
  - error handling S:143
  - files associated with S:11
  - opening, saving and closing S:17
  - red borders showing errors S:60
  - sample workflow for S:29
  - schemas S:3
  - tabs for, in a project S:21
  - testing S:49
  - view model editor S:20
- view --regenerate command C:273
- view --set-property command C:275
- view --update command C:279
- ViewLibrary category, contents of S:18
- views
  - adding generation schedules C:259
  - adding schedule exclusions C:262
  - configuring values for SQL parameters C:275
  - configuring with database sources C:263
  - configuring with data-service sources C:266
  - configuring with file sources C:267
- views (*continued*)
  - deleting C:272
  - deleting generation schedules C:272
  - listing generation schedules for C:274
  - obtaining information about C:274
  - regenerating C:273
  - removing old results C:273
  - setting run-as user C:279
  - showing dependencies C:272
  - specifying a style sheet engine C:265, C:268, C:271
  - updating C:279
  - See also generated views, SQL views, view generators, view models
- virtual database
  - about O:22
  - configuring attributes P:70
  - defined AD:359, API:93, C:317, O:71, P:299, S:185
  - displaying SQL views C:283
  - executing ad-hoc queries on C:282
  - schema
    - browsing catalogs P:64
    - browsing schemas P:64
    - browsing tables P:64
  - table cache size system property AD:144
  - types of schemas P:51, P:64
  - viewing and modifying ACLs P:72
- virtual database operations
  - about O:23, P:49
  - access permissions P:50
  - allowing creation of C:280
  - allowing groups to create P:67
  - allowing users to create P:65
  - creating C:87, P:50
  - defined AD:359, API:93, C:317, O:71, P:299, S:186
  - evicting from cache P:150
  - executing P:61
  - generating SQL views from P:60
  - listing in cache P:148
  - location in categories S:18
  - managing P:50
  - marking for scheduled caching P:139
  - modifying P:55
  - preventing creation of C:281
  - preventing groups from creating P:69
  - preventing users from creating P:68
  - qualified names for O:31
  - removing P:63
  - schemas for, generating P:57
  - SQL statements in C:90
  - tagging for on-demand caching P:146
  - unscheduling P:150
  - viewing P:55, P:57, P:59
  - viewing dependencies P:59
  - viewing details P:56
- virtual database service, configuring access permissions P:65
- virtualdatabase --allow-dbp-creation command C:280
- virtualdatabase --disallow-dbp-creation command C:281
- virtualdatabase --execute command C:282

virtualdatabase --show-tables command C:283  
 virtualeschema --deploy command C:285  
 virtualeschema --undeploy command C:286  
 virtual schema models, See metadata models

## W

### web browsers

requirements for Avaki software AD:5  
 setting for selecting run-as users P:27, P:54, P:92, P:224,  
 P:227

### web services API:2, API:8

about API:2  
 access permissions API:9  
 client examples API:9  
 data catalog API:18  
 data services API:34  
 database operations API:40  
 development framework API:5  
 document/literal API:3, API:5  
 provisioning, about P:205  
 rpc/encoded API:3, API:5  
 security API:8  
 with MIME API:3

### web services clients

defined AD:359, API:93, C:317, O:71, P:299, S:186  
 requirements for API:4

### web services description language, See WSDLs

### whoami command C:286

### whoami SOAP operation API:33

### wildcard characters in searches AD:235

### Windows

avoiding install pathnames with spaces AD:26  
 installing Avaki in AD:23  
 installing on Windows 2003 AD:22  
 requirements for AD:3  
 services, running under avaki local user account AD:11  
 shortcuts for AD:27  
 update for HTTP POST problem in web browsers AD:5  
 versions supported by Avaki O:16

### Windows domains

displaying for DGAS admission policies C:114  
 setting for DGAS admission policies C:111  
 unsetting for DGAS admission policies C:116

### Windows Services list C:5

### Workbench S:12

### workspace directory for Avaki Studio

described S:12  
 setting S:10

### WS API

accessibleDBOp API:42  
 accessibleDS API:36  
 accessiblePath API:19  
 authentication API:9  
 authorization API:9  
 chmod API:19  
 chown API:20  
 data access API:2

### WS API (continued)

data services SOAP operations API:34  
 executeAdHocDBOp API:43  
 executeAdHocDBOpWithOutput API:44  
 executeAdHocDBOpWithOutputAttach API:46  
 executeAdHocDBOpWithOutputString API:47  
 executeDBOp API:48  
 executeDBOpBytesInput API:49  
 executeDBOpGridFileInput API:50  
 executeDBOpWithOutput API:50  
 executeDBOpWithOutputAttach API:52  
 executeDBOpWithOutputString API:53  
 executeDS API:36  
 fileRead API:21  
 fileReadAttach API:21  
 fileReadString API:22  
 fileWrite API:23  
 getAttributes API:23  
 getDBOpOutput API:54  
 getDBOpOutputAttach API:55  
 getDBOpParameters API:56  
 getDBOpSchema API:56  
 getDBOpSchemaAttach API:57  
 getDBOpSchemaString API:58  
 getDSOutput API:37  
 getDSOutputAttach API:38  
 getDSOutputString API:38  
 getDSParameters API:39  
 getOutputString API:55  
 getSQL API:58  
 getSystemAttributes API:24  
 getUserAttributes API:24  
 grid server API:6  
 HTTP API:7, API:8  
 HTTPS API:7, API:8  
 isDSAvakiXML API:40  
 listDBConns API:59  
 listDBOps API:59  
 listDBOpsByDBConn API:60  
 listDomains API:25  
 listDSs API:40  
 listSearches API:25  
 ls API:26  
 lsSize API:26  
 mkdir API:27  
 mkdirParents API:27  
 mkdirParentsServer API:28  
 mkdirServer API:29  
 mv API:29  
 permissions API:30  
 ports API:6  
 removeAttribute API:31  
 rm API:31  
 search API:32  
 setAttribute API:32  
 tester API:33  
 whoami API:33

WS clients, See web services clients

WSDLs  
  about API:2  
  as SOAP contracts API:3  
  AvakiAPI.disco discovery file for .NET clients API:3  
  AvakiAPIIDocLit.wsdl API:3  
  AvakiAPIRpcEnc.wsdl API:3  
  AvakiAPIWithMIMEDocLit.wsdl API:3  
  AvakiAPIWithMIMERpcEnc.wsdl API:3  
  choosing API:5  
  document/literal API:3  
  editing API:6  
  locations of API:6  
  provided by Avaki API:11  
  rpc/encoded API:3  
  standards compliance API:1  
WSDLs directory O:35

## X

X Window System libraries required for Avaki install on  
  Unix AD:16  
XA drivers, configuring for database connectors C:69, P:7, S:36  
Xalan C:265, C:268, C:271, P:220, P:224, P:227  
XWorkHandler class for data services P:189  
XWorkUnit interface for data services P:189  
XML data in Avaki O:11, P:273  
XML indent size property AD:142  
XML schema  
  Avaki rowset  
    class-name element P:279  
    column-display-size element P:279  
    column-index element P:279

XML schema (*continued*)  
  core schema P:277  
  overview P:277  
  rowset-specific schema P:279  
  sample schema P:280  
data service  
  class element P:261  
  coherenceWindow element P:261  
  dataService element P:262  
  description element P:263  
  initParameter element P:263  
  inputParameter element P:264  
  inputSource element P:265  
  inputStream element P:266  
  isList element P:266  
  jarurl element P:267  
  logicBox element P:268  
  name element P:269  
  outputStream element P:269  
  ref element P:270  
  target element P:270  
  type element P:270  
  urlLogicBox element P:271  
  value element P:272  
  values element P:272

XSLT  
  in view generators, when not to use P:242  
  using in data service plug-ins P:180  
  See also Xalan, Saxon