Administration Guide **PUBLIC**

SAP Adaptive Server Enterprise 16.0 SP01
Document Version: 1.0 – December 11 2015

Clusters Users Guide

Content

1 How the SAP ASE Cluster Edition Works. 14

1.1 Integrated Clusterware. 15

1.2 The Cluster Coordinator. 17

1.3 The Quorum Device. 17

1.4 Database Devices. 18

1.5 Private Installation Versus Shared Installation. 19

1.6 Backup Servers. 19

1.7 Architectural Advantages of Clustered Environment. 20

1.8 Interconnected Networks in the Cluster. 20

Monitoring Links Between Instances. 21

1.9 Suggested Deployment Scenarios. 22

Configuring HA Failover for OLTP Applications. 23

Horizontal Scalability for DSS Reporting Applications. 24

Horizontal Scalability for OLTP Applications. 24

1.10 Client Technologies. 24

1.11 Support for Replication. 25

Configure Replication Agent. 25

Start Replication Agent. 26

2 Administer Clusters with sybcluster. 27

2.1 sybcluster and the SCC Agent. 27

Starting the SCC Agent for a Custer. 28

2.2 Starting sybcluster. 28

2.3 Using sybcluster. 29

Commands Active Before Connecting to a Cluster. 29

Commands Active After Connecting to a Cluster. 30

2.4 Connecting to the Cluster. 31

Authenticating the User. 31

Identifying the Agents. 32

2.5 Creating a Cluster. 33

2.6 Starting the Cluster. 33

2.7 Managing the Cluster. 34

Verifying a Cluster. 34

Displaying Cluster Information. 34

Changing Cluster Configuration Values. 36

Disconnecting from the Cluster. 38

Shutting the Cluster Down. 38

Dropping a Cluster. . . . 38
2.8 Managing an Instance. . . . 39
Displaying Information About the Instance. . . . 39
Adding an Instance. . . . 40
Verifying the Instance. . . . 40
Changing the Default Instance. . . . 41
Changing Instance Properties. . . . 41
Shutting an Instance Down. . . . 41
Dropping an Instance. . . . 42
2.9 Enabling sybcluster After Manually Creating the Cluster. . . . 42
2.10 Creating and Managing Auxiliary Servers. . . . 43
Creating Auxiliary Servers. . . . 43
Dropping Auxiliary Servers. . . . 44
Displaying Listening Port Information. . . . 44
Changing Listening Port Information. . . . 45
2.11 Upgrading the Server. . . . 45

3 Administer Custers with SAP Control Center for ASE. . . . 46

4 Use SAP ASE Cluster Edition with the Veritas Cluster Server. . . . 47
4.1 Features. . . . 47
4.2 Tasks. . . . 47
4.3 Components. . . . 48
4.4 Supported Platforms. . . . 49
4.5 Requirements and Limitations of Vcs Mode. . . . 50
4.6 Installing and Configuring the SAP ASE Cluster Edition on VCS. . . . 50
Installing the SAP ASE Cluster Edition. . . . 51
Creating a New Cluster for Storage Foundation Integration. . . . 52
Converting an Existing Cluster to Use SF for Sybase ASE CE. . . . 53
4.7 Managing a Cluster Under VCS Control. . . . 54
Starting and Stopping Instances. . . . 54
When to Add and Remove Instances. . . . 55
Increasing the Number of User Connections. . . . 55
Changing the Character Set or Sort Order. . . . 55
4.8 Membership Modes. . . . 56
Determining the Membership Mode. . . . 56
Changing Membership Mode. . . . 57
4.9 Understanding Failure Scenarios. . . . 57
4.10 VCS Troubleshooting. . . . 58
SAP ASE Cluster Edition Fails to Start. . . . 59
Veritas Log: "Sybase home directory does not exist". . . . 60
Instance Log: "Failed to bind to socket". . . . 60

Instance Log: "Membership service failed to receive initial message from Veritas cluster membership after 301 seconds. Exiting...". 60
Instance Log: "Failed to open quorum device 'device_path'. OS error 13, 'Permission denied'" 61
Instance Log: "basis_dsizecheck: attempt to open device 'device_path' failed, system error is: Permission denied". 62
Instance Log: "The configuration area in master device appears to be corrupt.". 62
Veritas Log: "Path not found". 63
VCS Shuts Down the Instance After It Starts and Issues Resource Faults. 63
VCS Cannot Shut Down the Instance Resource. 64
Finding Resource Faults for VCS Groups. 65
VCS Fails to Start. 66

5 Private Installation Mode. 68
5.1 Maintaining the Server Configuration Files. 68
Changing Configuration Options at Runtime. 69
Manually Changing the Configuration File. 69
If the Version Numbers Do Not Match at Start-Up. 70
Recommended Configurations. 70

6 Client Applications and Client/Server Interaction. 71
6.1 Open Client and OCS Version Support. 72
6.2 Failover in Client-Library Applications. 72
Enable Failover for Non-HA Applications. 72
Enable Failover in Client-Library Applications. 73
6.3 Extended High-Availability Failover. 73
Differences Between HA Failover and Failover in Clusters. 74
6.4 Using isql in a Clustered Environment. 74
6.5 Login Redirection. 75
Setting Connection Properties for Login Redirection. 76
6.6 Connection Migration. 76
Enabling and Disabling Connection Migration. 76
Difference Between Migration and Failover. 76
When Can Migration Occur?. 77
Criteria for Migration. 77
Migrated Contexts. 78
Client Migration Configuration Parameters. 78
6.7 Remote Procedure Calls in a Clustered Environment. 80
RPCs Where the Remote Server is a Cluster. 80
RPCs Where the Local Server is a Cluster. 80
RPCs Where Local and Remote Servers Are Instances in the Same Cluster. 81
sp_serveroption. 81
6.8 Reconnecting Clients When a Node Loses Power. 82

7 Using Backup Server in a Clustered Environment. 84
7.1 Nodes Joining Cluster During a Database or Transaction Dump. 84
7.2 Multiple Backup Servers. 84
Configuring SAP ASE Cluster Edition to Use Multiple Backup Servers. 85
Starting and Stopping Backup Servers. 87
Backing Up to Media. 87
Change Backup Servers Using Stored Procedures. 87

8 Run Job Scheduler in a Clustered Environment. 88
8.1 Installing and Configuring Job Scheduler 88
8.2 Running Job Scheduler in a Clustered Environment. 88
8.3 Shutting Down Job Scheduler. 88
8.4 Redirecting Scheduled Jobs. 89

9 Using Java in a Clustered Environment. 90

10 Using System Tables in the Cluster Edition. 91
10.1 timestamp Columns. 91
10.2 Changed Identity Values. 91
10.3 Controlling Fake-Table Materialization. 92

11 Monitoring Tables for Clusters 93
11.1 Configuring the System View. 93
11.2 Configuring Monitoring Tables. 94
11.3 Message Pipe Parameters 94
11.4 Changes for RPCs. 95
11.5 Monitoring Tables Specifically for the Cluster Edition. 95
11.6 Monitoring Tables that Return Identical Information for All Instances. 96
11.7 Monitoring Tables that Return Specific Instance Information. 96

12 Managing the Workload. 98
12.1 Logical Cluster Resources. 98
12.2 The System Logical Cluster. 98
12.3 Configuring a Logical Cluster. 99
Creating a Logical Cluster. 99
Adding Instances to a Logical Cluster. 100
Adding Routes to a Logical Cluster. 101
Starting a Logical Cluster. 102
12.4 Routing Rules. 102
12.5 Logical Cluster Attributes. 102
Open Logical Cluster. 103
System View Attribute. 104
Start-up Mode. 104

Down-Routing Mode. 105
Failover Mode. 106
fail_to_any Attribute. 106
Load Profile Attribute. 107
Login Distribution Mode. 107
Action Release. 108
Gather Mode. 108
Roles. 109
12.6 Configuring Failover 109
Adding Failover Resources. 110
12.7 Managing Logical Clusters. 111
User Tasks and Logical Clusters. 111
Workload Manager Thread. 111
Viewing Information about a Logical Cluster. 112
Creating and Dropping a Logical Cluster. 113
Adding Resources to a Logical Cluster. 114
Dropping Resources from a Logical Cluster. 114
Adding, Moving, and Dropping Routes. 114
Migrating Connections. 115
12.8 Administering Logical Cluster State. 116
Failover. 116
Failback. 116
Online. 117
Offline. 117
Deactivate. 117
Cluster and Instance States. 118
How States Change. 119
Asynchronous Commands and Logical Cluster States. 120
Using Action Descriptors. 121
Scheduling and Rescheduling a Failover. 122
12.9 Workload Distribution. 122
Workload Metrics. 122
Updating a User Metric. 123
Creating a User Metric. 123
Weighting the Workload Metrics. 124
Load Thresholds. 124
Load Profiles. 124
Login Redirection 125
Connection Migration. 125
12.10 Sample Load Profiles. 126
12.11 Creating and Configuring Your Own Load Profile. 127

Creating the Load Profile. 127
Building the Load Profile. 128
Associating the Load Profile with a Logical Cluster. 129
Changing the Load Profile. 129
12.12 Troubleshooting. 129

13 I/O Fencing. 131
13.1 Setting Up I/O Fencing. 131
Setting Up I/O Fencing on Linux. 132

14 Quorum Device Heartbeat. 133
14.1 Configuring the Quorum Device Heartbeat. 133

15 Distributed Checkpoints. 135
15.1 Cluster Checkpoints. 135
Execution of a Distributed Checkpoint. 135

16 Distributed Transactions on a Shared-Disk Cluster. 136
16.1 Using DTM on the Shared-Disk Cluster. 136
16.2 The Cluster as the Resource Manager. 136
16.3 Handling Requests on Nonowner Instances. 137
16.4 Handling Instance Failures 137
16.5 Transaction Coordination With ASTC. 138
16.6 The Impact of Connection Migration. 138
Connection Migration and Performance. 138
When Connections Cannot Migrate. 139
16.7 Configuration and System Issues. 140

17 Recovery. 142
17.1 Recovery Process. 142
17.2 Single Transaction Log 143
Understanding Distributed Timestamps. 143

18 Instance Recovery. 145
18.1 Single-Instance Recovery. 145
Single Transaction Log 145
18.2 Multiple Simultaneous Failures 146
Enabling Multiple Simultaneous Failovers. 146
18.3 Recovery Algorithm. 147

19 Single Instance Databases. 149
19.1 Binding Single-Instance Databases to Single-Instance Logical Clusters. 149
19.2 Configure a Cluster for a Single-Instance Database. 150
19.3 Administering Single-Instance Databases. 152

Remove a Binding. . . . 152
Migrate a Single-User Database to Another Node. . . . 152
Migrate the Primary Instance. . . . 153
Remove Single-Instance Databases. . . . 153
Convert a Regular Cluster Database to Single-Instance Database. . . . 154
Converting a Single-Instance Database to a Regular Cluster Database. . . . 154
19.4 Limitations. . . . 154

20 Temporary Databases. . . . 156
20.1 Local Temporary Databases. . . . 156
Local User Temporary Databases. . . . 156
Local System Temporary Databases. . . . 157
20.2 Global Temporary Databases. . . . 158
20.3 Characteristics of Temporary Databases. . . . 159
20.4 Creating Temporary Databases. . . . 159
Creating Local System Temporary Databases. . . . 160
Creating Local and Global User Temporary Databases. . . . 160
20.5 Binding Users and Applications to Temporary Databases. . . . 161
Creating and Managing Temporary Database Groups. . . . 161
What You Can Bind. . . . 162
How the Session Binding is Determined. . . . 162
Creating and Managing Bindings. . . . 163
Displaying Group and Binding Information. . . . 163
Dropping Temporary Databases. . . . 164
20.6 Restrictions for Temporary Databases. . . . 165
20.7 Private Device Support for Local Databases. . . . 166
Using Private Devices for Temporary Data. . . . 166
Creating Private Devices Using disk init. . . . 167
Reinitializing Private Devices Using disk reinit. . . . 167
Dropping Private Devices Using sp_dropdevice. . . . 168
Displaying Private Device Information Using sp_helpdevice. . . . 168
Using create database and alter database with a Private Device. . . . 169
disk refit. . . . 170
20.8 create database. . . . 171
20.9 sp_tempdb. . . . 173
20.10 sys_tempdbid. . . . 175
20.11 tempdb_id. . . . 176

21 Adding Space to an Archive Database. . . . 178

22 Memory in a Clustered Environment. . . . 179

23 Cluster Caches. . . . 180

23.1 Global Caches. . . . 180
23.2 Local Caches. . . . 180
23.3 Creating and Configuring Named Data Caches. . . . 181
sp_cacheconfig. . . . 181
23.4 Configuring and Using Multiple Buffer Pools. . . . 185
sp_poolconfig. . . . 185
23.5 Binding Objects to Named Caches. . . . 188
sp_bindcache. . . . 188
Getting Information About Bound Caches. . . . 189
sp_unbindcache and sp_bindcache_all. . . . 190
23.6 Configuration File. . . . 191
Format of a Local Named Cache. . . . 191
Extra Line in Local Cache Entries. . . . 192
Deleted Named Cache with Global Configuration. . . . 193
Named Cache with Local Configuration. . . . 193
Deleted Entries with Valid Configuration. . . . 193
Consequences of Creating a Local Configuration in the Presence of a Global Configuration . . . 194
23.7 Limitations. . . . 195

24 Locks in a Clustered Environment. . . . 196
24.1 Deadlocks. . . . 196
24.2 Retention Locks. . . . 196
24.3 Object Coherency for Metadata and Global Variables. . . . 197
24.4 Cluster Lock Requests and Task Request Status. . . . 197
24.5 Enabling Lock Remastering. . . . 197
24.6 Preventing Out-of-Lock Situations. . . . 198
24.7 Lock Management Enhancements. . . . 199
24.8 Improved Concurrency for Partition-Level Online Operations. . . . 200

25 Thresholds in a Clustered Environment. . . . 201
25.1 dbcc thresholds Output. . . . 201
25.2 dbcc dbtable Output. . . . 202
25.3 dbcc dbrepair with remap Option. . . . 202
25.4 dbcc dbrepair with newthreshold Option. . . . 203

26 Cluster Inter-process Communication . . . 204
26.1 Remote Directory Memory Access (RDMA). . . . 204

27 Using InfiniBand. . . . 206
27.1 Setting the Buffer Space. . . . 206
Configuring Buffer Space on Linux. . . . 206
27.2 Configuring InfiniBand in a Cluster. . . . 206

28 Support for mount and unmount Commands. 207

29 Using sp_showplan. 208

30 Restrictions for Using Encrypted Columns in a Clustered Environment. 209

31 Adding SSL Common Name. 210

31.1 Specifying a Common Name with sp_listener. 211

32 Removing Column Required for the Search Argument from the Substitution List. 212

33 Performing Lazy Decompression Using a RID Join. 213

34 Configure the Operating System 214

34.1 Using the stty Setting. 214

34.2 Restoring Correct Permissions. 214

34.3 File Descriptors and User Connections. 214

Displaying Current Soft and Hard Limits. 215

Increasing the Soft Limit. 215

Increasing the Hard Limit. 216

34.4 Adjust the Client Connection Timeout Period. 217

Adjusting the Client Connection Timeout. 218

34.5 Check for Hardware Errors. 218

34.6 Monitor Operating System Resources 218

34.7 Check Database Integrity. 219

35 Set Up Communications Across the Network. 221

35.1 How SAP ASE Determines Which Directory Service Entry to Use. 221

35.2 How a Client Uses Directory Services. 222

35.3 Creating a Directory Services Entry. 222

35.4 Supported Directory Drivers. 222

35.5 Contents of the interfaces File. 223

35.6 interfaces File Format. 224

Components of an interfaces File Entry. 225

35.7 Create a Master interfaces File. 227

Using dsedit or dscp to Create a Master interfaces File. 228

Using a Text Editor to Create a Master interfaces File. 228

35.8 Configure the interfaces File for Multiple Networks. 229

Configuring the Server for Multiple Network Handlers. 229

Configure Client Connections. 229

Configuring for Query Port Backup. 231

35.9 IPv6 Technology 232

IPv6 Infrastructure. 232

36 Troubleshoot. 235

36.1 Verifying the Cluster Environment. . . . 235
36.2 Restarting the Cluster Using a dataserver Binary from an Earlier Version of SAP ASE. . . . 236
36.3 Errors Accessing Disk Devices. . . . 237
36.4 Verifying That the Cluster Is Down. . . . 238
36.5 Creating Cluster Using sybcluster Fails With Error -131. . . . 238
36.6 Cluster Creation Fails Leaving Files in $SYBASE Directory. . . . 239
36.7 SCC Agent Starts but sybcluster connect Fails. . . . 239
36.8 Disk Devices in Use on Linux. . . . 240
36.9 Instances Fail to Join the Cluster. . . . 240
36.10 Verifying RDMA Support. . . . 241
36.11 SAP ASE Boot Errors with RDMA. . . . 242
36.12 Private Interconnect Failure. . . . 243
36.13 Client Connection Failover Fails. . . . 243
36.14 Client Fails to Reconnect to Alternate High-Availability Servers. . . . 244
36.15 sybcluster Cannot Connect If All Connections Use SSL. . . . 244
36.16 jConnect Sample Disables HA. . . . 244
36.17 PC-Client Installation – java.lang.NoClassDefFound Error. . . . 245
36.18 The Cluster Entry "name" Did Not Contain Any Servers. . . . 245
36.19 After Password Change, sybcluster Cannot Manage the Cluster. . . . 246
36.20 Agent "cannot be found". . . . 246
36.21 SCC Plug-In Register Error. . . . 247
36.22 Data On Disk Unavailable: Problems Affecting Database Creation. . . . 248
36.23 sybcluster Cannot Find interfaces File. . . . 248
36.24 When the Server Fails to Start. . . . 248
36.25 Investigating the Port Assignment. . . . 249
36.26 If a Stale Server Process Is Retaining Use of the Port Number. . . . 250
36.27 Error when Executing an ESP. . . . 250

37 Use the Lightweight Directory Access Protocol as a Directory Service. . . . 251
37.1 LDAP Directory Services Versus the SAP Interfaces File. . . . 252
37.2 The libtcl*.cfg File. . . . 254
37.3 Enabling LDAP Directory Services. . . . 255
37.4 Keywords for the LDAP URL Variables. . . . 256
37.5 Adding a Server to the Directory Services. . . . 256
37.6 Multiple Directory Services. . . . 257
37.7 Encrypting the Password. . . . 258
37.8 Performance When Using LDAP. . . . 258
37.9 Migrating from the Interfaces File to LDAP. . . . 259
37.10 Using LDAP Directory Services with the Shared-Disk Cluster. . . . 260

38 Localization Support. . . . 262
38.1 Language Modules. . . . 262

38.2 Default Character Sets . . . 263
38.3 Supported Character Sets. . . . 264
Arabic Character Sets. . . . 264
Baltic Character Set. . . . 264
Simplified Chinese Character Sets. . . . 264
Traditional Chinese Character Set. . . . 265
Cyrillic Character Sets. . . . 265
Eastern European Character Sets. . . . 266
Greek Character Sets. . . . 266
Hebrew Character Sets. . . . 266
Japanese Character Sets. . . . 267
Korean Character Set. . . . 267
Thai Character Sets. . . . 267
Turkish Character Sets. . . . 268
Unicode Character Set. . . . 268
Vietnamese Character Set. . . . 268
Western European Character Sets. . . . 269
38.4 Character Set Names. . . . 269
38.5 Load Character Sets . . . 271
38.6 Language Definition Files Sort Order . . . 273
38.7 Character Set Conversion. . . . 275
Conversions Between Server and Client. . . . 275
38.8 Sort Orders. . . . 276
Available Sort Orders for Character Sets. . . . 276
Default Unicode Sort Orders. . . . 278
38.9 Language Modules. . . . 279
Installing a New Language Module. . . . 280
Message Languages. . . . 280
38.10 Localization Directories. . . . 280
About the locales Directory. . . . 281
About the charsets Directory. . . . 281
Format of locales.dat File Entries. . . . 281
How Client Applications Use locales.dat. . . . 282
38.11 Changing Localization Configuration . . . 282
Configuring SAP ASE for Other Character Sets. . . . 283
Editing the locales.dat File. . . . 284
Changing Backup Server Localization Configuration. . . . 285

39 SAP ASE Error Logging. . . . 286
39.1 Set Error Log Paths. . . . 286
Change Error Log Path. . . . 287
39.2 Manage Messages. . . . 287

Log User-Defined Messages. 287
Log Auditing Events. 288

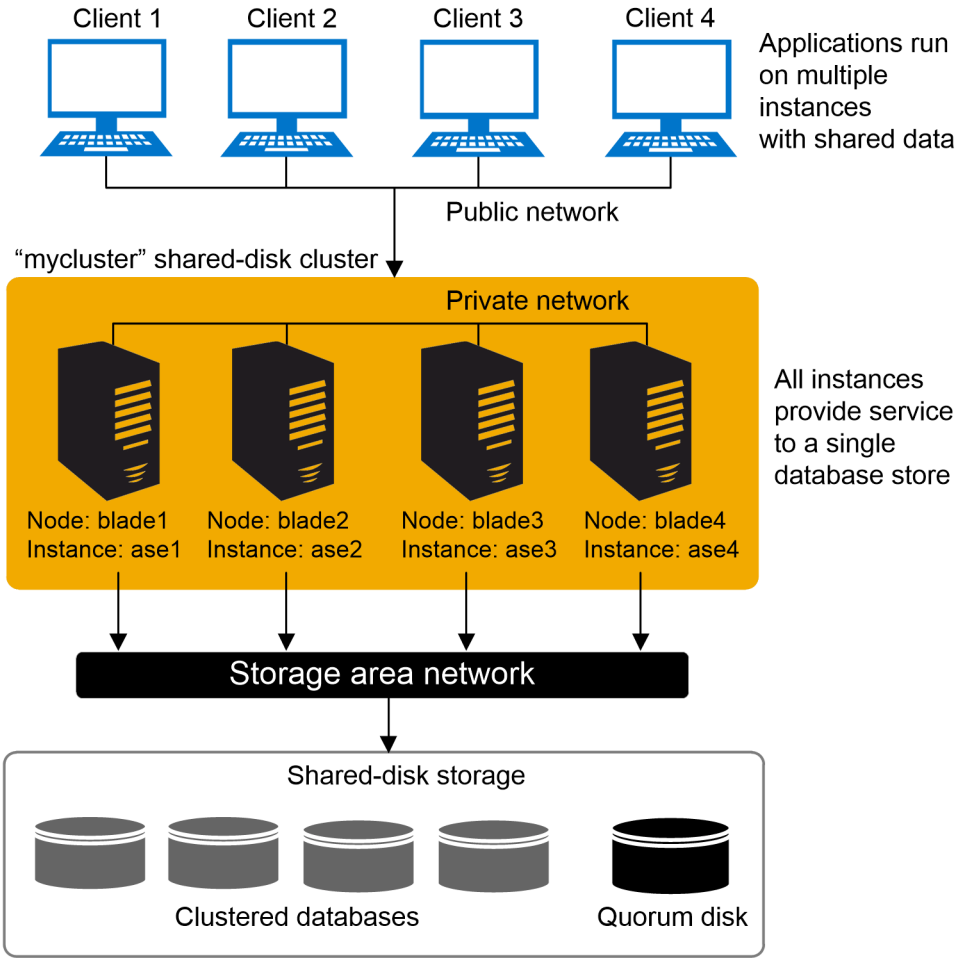
40 Adding Auditing 289
40.1 Audit System Devices and Databases. 289
40.2 Running auditinit with the SAP ASE Cluster Edition 290
40.3 Preinstallation Tasks for Auditing Devices. 290
40.4 Installing Auditing. 291
Configuring the SAP ASE Cluster Edition for Auditing. 291
Creating a Device for an Audit Table 292
Creating a Device for the Audit Database Transaction Log. 293
Enabling Auditing. 295

41 Install Online Help for Transact-SQL Syntax 297
41.1 Default Device for the sybsyntax Database. 297
41.2 Installing sybsyntax. 298

1 How the SAP ASE Cluster Edition Works

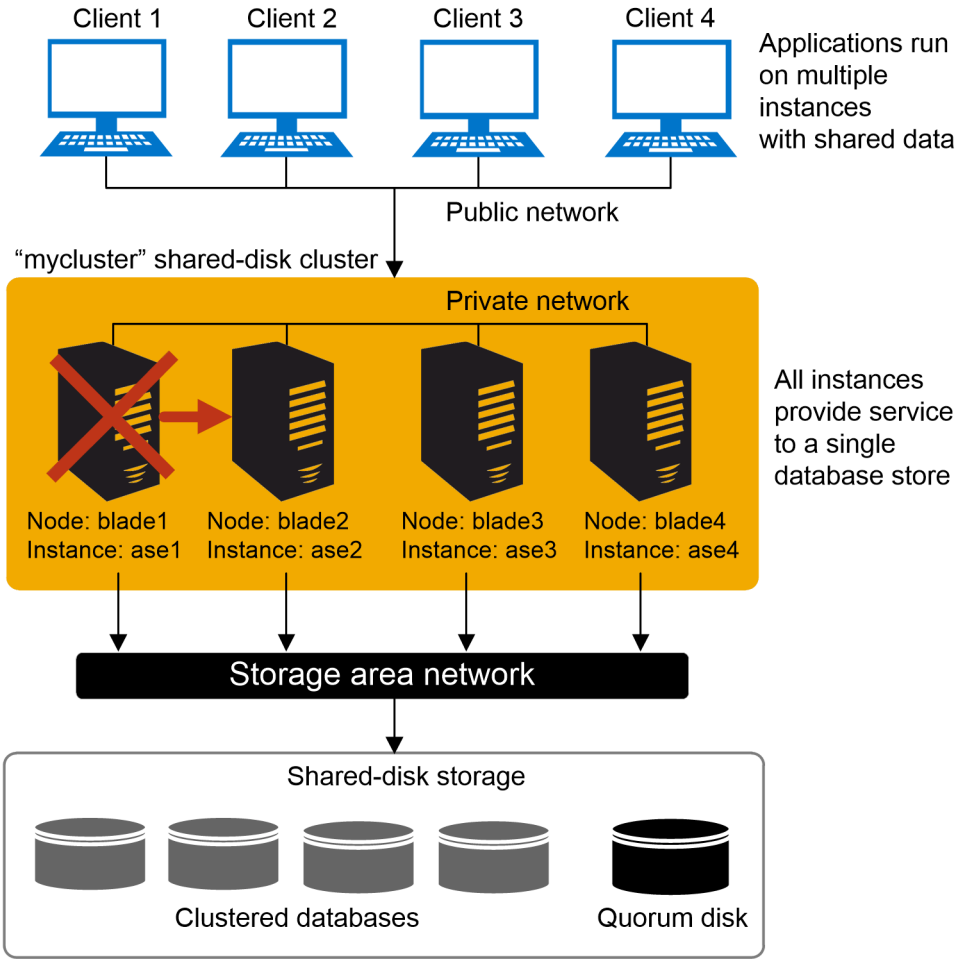
The SAP Adaptive Server Enterprise Cluster Edition (SAP ASE Cluster Edition) allows you to configure multiple server instances to run as a shared-disk cluster.

In a clustered environment, multiple nodes (machines running server instances) communicate with each other, making all data accessible from any instance. Connected instances form a cluster, which allows the instances to work as a single system that manages a single set of databases on shared disks. A clustered environment allows servers to scale and provide highly available services to clients. A single server process ID (SPID) identifies a cluster and all instances in the cluster.



This figure shows a clustered system, where clients connect to a shared-disk cluster named "mycluster", which includes the "ase1", "ase2", "ase3", and "ase4" instances running on nodes "blade1", "blade2", "blade3", and "blade4".

If one cluster member (an instance on a node) fails, its workload transfers to surviving cluster members, as shown in this figure.

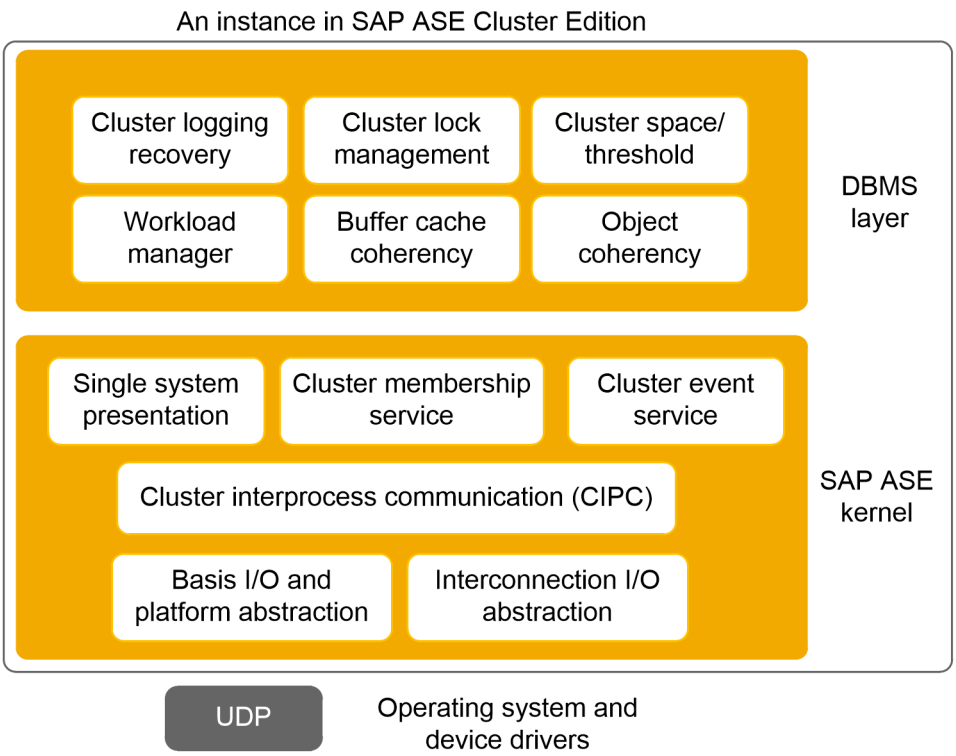


> **Note**
>
> The SAP ASE Cluster Edition can handle multiple failures if they do not happen concurrently, and can recover fully from the initial failure before a subsequent failure occurs.

1.1 Integrated Clusterware

All clusterware is integrated directly into the SAP ASE Cluster Edition. No external clusterware is required.

The following figure illustrates cluster infrastructure components. Some of these components are native to the SAP ASE Cluster Edition; others are cluster-aware extensions of the existing SAP ASE infrastructure.



The SAP ASE Kernel

Native cluster infrastructure components include:

- Cluster membership service – manages cluster membership, and detects and handles instance failure.
- Cluster interprocess communication (CIPC) – provides messaging services and an interconnection abstraction layer that allows the instances to communicate with each other via redundant pathways.
- Cluster event service – supports a generic event-publishing and subscription mechanism for cluster-wide events.

DBMS Layer

These key components in the SAP ASE DBMS layer are extended to work in the SAP ASE Cluster Edition environment:

- Buffer cache coherency – handles coherency issues related to the shared buffer cache and supports cache-to-cache transfer for allocation pages, index pages, data pages, object allocation map (OAM), and global allocation map (GAM) pages.
- Cluster lock manager – supports distributed locking for coherency control across the cluster.
- Cluster logging and recovery – handles logging from all instances, and fail-over database recovery.

- Cluster space and threshold – handles space and threshold management in a distributed environment.
- Object coherency – handles coherency issues related to sharing and transferring metadata and global variables. Object coherency must serialize updates to shared objects and make the latest changes available to all instances in the cluster.
- Workload manager – a server module that provides application-level management of resource allocation, availability, and load distribution.

1.2 The Cluster Coordinator

The cluster coordinator handles specific tasks related to membership management and recovery. Any instance attempting to join an existing cluster first contacts the cluster coordinator.

There are no start-up parameters to indicate that a particular instance is the cluster coordinator; configure it just like any other instance in the cluster. Initially, the cluster coordinator is the first instance you start. If the cluster coordinator exits, another instance dynamically assumes the coordinator role.

1.3 The Quorum Device

The quorum device includes configuration information for the cluster, and is shared by all cluster members. The quorum device must be on a raw partition and be accessible to all nodes that host cluster instances.

The SAP ASE Cluster Edition uses the quorum device as:

- A location to perform cluster membership management, including voting and arbitration for members joining.
- A persistent place to store configuration data used by instances and the agent.
- A communications medium and synchronization point.

The quorum device includes information about:

- The name of the cluster, the number of instances in the cluster, the path to the directories containing the interfaces file, log files, master device, and other required configuration information.
- Cluster view records that indicate the state (up or down) of each instance in the cluster.
- The area that SAP ASE Cluster Edition uses to determine the proper cluster membership when an instance failure is detected.

Create the quorum device when you configure the cluster (see the installation guide for your platform). After the initial configuration, use `sybcluster` or the `qrmutil` utility to back up, restore, and reconfigure the quorum device.

For more information about `sybcluster` and `qrmutil`, see the *SAP ASE Utility Guide*.

1.4 Database Devices

In the SAP ASE Cluster Edition, database devices – except those private devices that are used by local user temporary databases – must be raw devices (also known as character devices).

You cannot use block devices as database devices because they can buffer disk writes at individual hardware nodes, causing data inconsistency among cluster instances. However, you can create private devices on block devices. Private devices are used only by local user temporary databases. For more information about configuring raw devices, see your operating system documentation.

You can create local user temporary databases on private devices, but you must create local system temporary databases on shared devices. For the SAP ASE Cluster Edition, you can use less expensive, local file system devices (block devices) for managing the storage needs of temporary data in the cluster. These devices are added as private devices and can only be used by local user temporary databases.

Linux

On Linux systems, the path `/dev/sda` is a block device and should not be used. However, you can bind this block device to a raw device, such as `/dev/raw/raw1`.

You can distinguish character (raw) devices with the file type displayed using `ls -l` command. Block devices include a `b` as the file type, and character (raw) devices have a `c` as the file type:

```
[joeadminitrator@isles ~]$ ls -l /dev/sda
brw-rw----  1 root disk 8, 0 Nov 29 06:15 /dev/sda
[joeadministrator@isles ~]$ ls -l /dev/raw/raw1
crw-------  1 sybase sybase 162, 1 Nov 29 12:17 /dev/raw/raw1
```

Database devices in the SAP ASE Cluster Edition must support SCSI-3 persistent group reservations (SCSI PGRs). The SAP ASE Cluster Edition uses SCSI PGRs to guarantee data consistency during cluster membership changes. We cannot guarantee data consistency on disk subsystems that do not support SCSI PGRs (you may use this configuration for test and development environments where data corruption is tolerated).

PGRs are a feature of the SCSI-3 protocol. However, many storage area networks (SANs) that employ less expensive SATA disks still provide this functionality. Contact your storage vendor to verify that your system supports SCSI-3 PGRs.

For more information about devices and I/O fencing, see the installation guide for your platform.

> **Note**
>
> I/O fencing is not supported on Solaris non-global zones.

Related Information

Temporary Databases [page 156]

1.5 Private Installation Versus Shared Installation

You can configure your cluster using a shared or private installation.

The requirements for each type of installation are:

- Shared installation – requires a shared file system that is created using a Network File System (NFS), or a cluster file system. A cluster that is created using a shared installation supports a single `$SYBASE` installation directory, SAP ASE home directory, and server configuration file.
- Private installation – supports a separate `$SYBASE` installation directory, SAP ASE home directory, and server configuration file for each instance. Parity among the server configuration files is maintained by the master configuration file on the quorum device.

> **Note**
>
> Use LDAP as your directory service when using a private installation.

The cluster input file contains an entry that identifies whether the installation mode is shared or private:

```
installation mode = private | shared
```

To set up a private or shared installation in `sybcluster`, see the installation guide for your platform.

Related Information

Private Installation Mode [page 68]

1.6 Backup Servers

You can configure multiple Backup Servers to perform dump and load operations.

`dump` and `load` commands are routed to the appropriate Backup Server depending on how you configure the cluster:

- Dedicated – each instance in the cluster is assigned a specific Backup Server.
- Round robin – instances are not assigned a specific Backup Server, but when they perform a dump or load, the cluster assigns a Backup Server for the job according to availability.
- A single Backup Server (named SYB_BACKUP) for the cluster.

Related Information

Using Backup Server in a Clustered Environment [page 84]

1.7 Architectural Advantages of Clustered Environment

The SAP ASE Cluster Edition allows servers to work in a shared-disk environment. You can group multiple servers to provide a single-system view of shared databases that delivers improved reliability and ease of management.

A group of loosely coupled servers, each of which can function as a nonclustered server, work together to provide the user with a single database system image. The servers in the cluster jointly manage a single installation of databases residing on shared-disk devices.

The primary advantages of the SAP ASE Cluster Edition architecture are:

- Improved availability – the shared-disk nature of the architecture means that applications can continue to run as long as a single cluster member remains viable, even after several others fail.
- Simple management – does not require data repartition as cluster membership changes, because data is shared by all instances.

> **Note**
>
> The SAP ASE Cluster Edition provides a distributed architecture. Internode communication takes place via a network interconnection, not, as with nonclustered SAP ASE, via shared memory (high-speed system memory). Applications that minimize internode messaging yield the best performance in the SAP ASE Cluster Edition environment.

1.8 Interconnected Networks in the Cluster

You can configure one or two interconnected networks between instances in the cluster. Although one interconnected network is sufficient, two interconnected networks allow for a redundant and more robust cluster.

> **Note**
>
> Secondary interconnected networks are not supported with remote directory memory access (RDMA).

The interconnected networks form a series of logical links between every instance in the cluster. These links send messages between instances and are monitored by the SAP ASE Cluster Edition. If any failures are detected, the SAP ASE Cluster Edition reroutes traffic over the alternate networks between the instances.

An interconnection link may fail for various reasons, such as a physical failure due to a cable that is disconnected or broken, a power supply failure from a piece of network infrastructure equipment, or a software failure within the networking stack. The SAP ASE Cluster Edition detects these failures by monitoring the traffic flow between instances. Each instance monitors the messages sent over the various links. A link is considered operational as long as there are incoming messages.

An instance may not always send messages because the node has failed or the link is down. If a particular instance in the cluster is quiet, the SAP ASE Cluster Edition initiates an active probing mechanism to determine if the node supporting the instance can be contacted using the network link. This mechanism ensures that periods of inactivity do not falsely trigger a link failure and subsequent switching event.

If a link is flagged as inoperable, periodic attempts are made to establish whether the link has been restored, so normal operations can be resumed without manual intervention.

> **Note**
>
> Link monitoring is automatically implemented when more than one network is defined in the SAP ASE Cluster Edition.

1.8.1 Monitoring Links Between Instances

The `monCIPCLinks` monitoring table monitors the state of the links between instances in the cluster. `monCIPCLinks` includes two states for each link: "passive" and "active".

A logical cluster and each instance in the cluster can have different states:

- A logical cluster has an overall (global) state that determines, for example, whether the cluster is offline or online.
- Each instance has a state that describes the state of that particular instance as it is perceived by a logical cluster.

The "passive" state is used to monitor day-to-day messages sent over the links. The SAP ASE Cluster Edition gathers the active state when there is no message traffic over the link. The status of the link is described as a "state", and each state has an age associated with it, in milliseconds. The states include "Up", "Down", and "In doubt". The state is "In doubt" when messages are not sent between the instances.

When the cluster is healthy, regular internode traffic is used to determine the state of the link. This is referred to as passive monitoring, and maintains the link's "passive" state. If the monitoring that determines the state occurs during a period of inactivity in the cluster, the defined state may become stale and unreliable (that is, a state that is determined to be "Up" during a period of inactivity may in fact be "Down", but the inactivity prevents the `monCIPCLinks` table from showing this in the result set). This inactive state is described in the `PassiveState` column as "In doubt." Once a link is marked as "In doubt", the active link state monitoring it is triggered and the value described by the `ActiveState` column is valid.

Each of the active and passive states has an age associated with it that shows when the state was last updated. If the normal traffic is sufficient to maintain the link state, the active state is not updated, and its associated age value becomes large. The large value indicates that the associated state may no longer accurately represent the true state of the link.

If instances are not sending messages, the `PassiveState` is listed as "In doubt", but the `ActiveState` shows the actual state: "Up", "In doubt", or "Down".

This example shows a two-node cluster in which both links are running and have traffic flowing between them. Because the `PassiveStateAge` is 0 for all links, you can assume the output is a true reflection of the link state:

```
InstanceID    LocalInterface     RemoteInterface         PassiveState
      PassiveStateAge    ActiveState    ActiveStateAge
------------   --------------     ------------            --------------
      --------------     ------------   ---------------
2              ase2               ase2                    Up
      0                  Up             10300
2              blade2             blade1                  Up
      0                  Up             0
```

```
1              ase1                ase2                    Up
      0                   In doubt       179001
1              blade1              blade2                  Up
      0                   Up             100
```

This example shows the same two-node cluster after the primary interconnected network fails. The `PassiveState` value for the link between the network endpoints "ase1" and "ase2" is "In doubt", and the value for the `PassiveStateAge` is "large" (indicating that the `ActiveState` represents the true state of the links). The `ActiveState` value is younger and shows the links as "Down":

```
InstanceID    LocalInterface     RemoteInterface         PassiveState
      PassiveStateAge    ActiveState    ActiveStateAge
------------   --------------     ------------            --------------
      --------------     ------------   ---------------
2              ase2                ase1                    In doubt
      13500              Down           700
2              blade2              blade1                  Up
      0                  Up             700
1              ase1                ase2                    In doubt
      13600              Down           400
1              blade1              blade2                  Up
      0                   Up             400
```

> **Note**
>
> There is a slight delay between the failure of a link and the time the active state truly reflects the state of the link.

Ignore any state with the value "large" for `ActiveStateAge`, since this indicates that the link is old, and its value may be inaccurate. When the link state is old and the value for `ActiveStateAge` is "large", active monitoring is triggered by the absence of messages, but it has not yet determined the link state.

> **Note**
>
> When you set up both primary and secondary interconnected networks in your cluster input file, do not restart the cluster unless both interconnected networks are running.

Related Information

Cluster and Instance States [page 118]

1.9 Suggested Deployment Scenarios

The SAP ASE Cluster Edition supports high availability failover (HA failover) for online transaction processing (OLTP) applications, and horizontal scalability for data distribution service (DDS) reporting applications and OLTP applications. Consider the following about clustered database architectures:

- Increased availability – if a node fails, other nodes in the cluster continue to run, and the database continues to be available.

- Increased manageability – multiple applications and databases can be consolidated into a single cluster, thereby reducing management complexity and introducing economies of scale.
- Reduced total cost of ownership – software that can be deployed on industry-standard, nonproprietary hardware, thereby reducing the cost of purchase, maintenance, and support.

1.9.1 Configuring HA Failover for OLTP Applications

If you are running a high availability (HA) production environment using either the capabilities provided by SAP Failover in an active-passive (standby) or active-active (companion) configuration, or an operating system-provided clusterware (such as Sun Cluster or Veritas Cluster), replace them with one of the SAP ASE Cluster Edition scenarios.

Choose and configure a failover scenario that best represents what your company would want given the service-level agreement requirements of the application and financial constraints.

1:1 (active-passive)	Cluster nodes and instances are set up in pairs with an idle (passive) node and instance that wait for the corresponding active node to fail. This scenario is cost effective only in extreme environments where requirements do not tolerate service degradation in any failover scenario, including multiple failures.
1:1 (active-active)	Cluster nodes and instances are set up in pairs, and each pair services a separate application and database (instance) while monitoring each other in “companion mode” in case the other fails. Although this scenario mimics the current HA option and provides full utilization of resources, service levels during failed-over processing degrades unless you maintain resource capacity during normal processing that is low enough (less than 50 percent) to provide sufficient capacity for a single instance to run the workload of both.
N:1 (N active nodes covered by a single passive standby node)	This scenario provides a single passive standby node and instance to monitor an arbitrary number of active instances. While this is the most cost-effective scenario, multiple node failures that fail instances over to a single node can lead to a level of service degradation that can make this scenario unacceptable. Configure the passive node with additional capacity (for example, CPUs and memory) to mitigate the degradation in the most likely failover scenarios.
N:M (N active nodes and instances covered by M passive standby nodes)	This model provides an arbitrary number (M) of passive standby instances to monitor an arbitrary number (N) of active instances. This option reduces costs while covering most typical failure scenarios. The choice of the number of passive standby nodes is typically driven by cost, system-level mean time between failures (MTBF) statistics, and the business impact of running at degraded service levels in a multiple-node failure scenario. In most failure scenarios, you can configure passive nodes with additional capacity to mitigate this degradation.

1.9.2 Horizontal Scalability for DSS Reporting Applications

If you have applications in which the online activity consists primarily of large-scale reports and decision support systems (DSS) querying from a large user population, consider creating a multiple-node cluster in which all server instances service these same applications.

When organizing this system, pay attention to:

- The scalability of users, queries, and response time as the workload expands from one to `<N>` nodes.
- The load distribution of clients across instances, and their relationship to instance capacity and performance.

1.9.3 Horizontal Scalability for OLTP Applications

Scale online transaction processing (OLTP) workloads on a shared-disk cluster by separating the data to different databases.

Applications that are read- and write-intensive – such as OLTP applications – traditionally pose challenges for scalability because of resource contention while maintaining the ACID properties of a database system. The horizontal scalability of OLTP applications adds additional challenges because of the data coherency and messaging that is required across the server instances in a shared-disk cluster. The physics of computers and networking does not allow OLTP applications to scale infinitely on a shared-disk cluster. As the number of nodes and the load and number of users increase, the amount of internode messaging necessary to maintain buffer coherency across the nodes increases exponentially.

You can scale OLTP workloads on a shared-disk cluster by partitioning the application and data into mutually exclusive sets (separating the data to different databases) to avoid processing coordination across server instances and access the data for an application from the same instance. Because of this, you must carefully consider how you partition "data" at a database level to eliminate log and data contention across the participating instances. You can facilitate access to this segmented "data" through a single instance with logical clustering and workload management.

Logical clusters allow you to allocate distinct instances to different applications or workloads, logically enabling these groups of data to operate under a single cluster. This reduces inter-instance access and, combined with dedicated temporary databases for each of the instances, helps to deploy an OLTP application in the SAP ASE Cluster Edition that supports continuous data availability.

1.10 Client Technologies

The SAP ASE Cluster Edition supports a single-system presentation, where multiple instances that make up the cluster appear to clients as a single system. Client technologies allow clients to connect logically to a cluster while physically connected to individual instances.

This logical connection allows SAP ASE to redirect the client to various instances in the cluster and to dynamically provide the client with high-availability fail over data.

The client technologies include:

- Login redirection – when the client reconnects to another instance in the cluster.
- Connection migration – when an established connection moves to another instance in the cluster.
- Extended high-availability failover – allows the client to fail over multiple times until it finds the first available or least-loaded instance.

Related Information

Client Applications and Client/Server Interaction [page 71]

1.11 Support for Replication

The SAP ASE Cluster Edition supports replication using Replication Server and the Replication Agent (RepAgent) thread.

A clustered database can be a source or a destination in a clustered system. You can perform all of the tasks, such as configuring RepAgent or marking tables for replication, from any instance in the cluster. Replication status is coherent across the entire cluster.

> **Note**
>
> The SAP ASE Cluster Edition does not support SAP ASE Replicator, which requires the unsupported `dbcc logtransfer` interface.

1.11.1 Configure Replication Agent

When you configure a primary database in a clustered system, the server name that you specify should be the cluster name.

Procedure

Display the cluster name using `select @@servername`.

The syntax for `sp_config_rep_agent` does not require a cluster or instance name. By default, both the SAP ASE Cluster Edition and the nonclustered SAP ASE assume the value of `select @@servername`. In the SAP ASE Cluster Edition, this statement returns the current cluster name:

```
1> select @@servername
2> go
--------------------------
```

```
MYCLUSTER
```

1.11.2 Start Replication Agent

By default, RepAgent starts on the coordinator. However, you can configure RepAgent to start on any instance in the cluster.

Procedure

1. Configure RepAgent on the primary database `pdb` to always start on a specific instance, for example "ase2":

   ```
   sp_config_rep_agent pdb, "cluster instance name", "ase2"
   ```

2. After configuration, restart RepAgent using `sp_start_rep_agent` for the new configuration to take effect.
3. Return to the default behavior with RepAgent always starting on the coordinator:

   ```
   sp_config_rep_agent pdb, "cluster instance name", "coordinator"
   ```

Results

When an instance starts on its node, it checks if databases are configured to start on its node. If yes, and if the database is marked to start automatically, the RepAgent starts.

When the coordinator starts, it starts all RepAgents not configured to start on a specific instance. If the coordinator fails, or is stopped with a graceful shutdown, a RepAgent starts on the new coordinator.

If a RepAgent is configured to start on a specific instance other than the coordinator, and this instance is shut down or fails, the RepAgent starts on the coordinator.

2 Administer Clusters with sybcluster

The interactive `sybcluster` utility allows you to perform administrative tasks for the cluster from the command line.

Use `sybcluster` to create a cluster, add an instance, start and stop a cluster or instance, display status information, and so on. Alternatively, you can perform these tasks using the SAP Control Center. See Administer Custers with SAP Control Center for ASE [page 46].

You cannot manage temporary databases or logical clusters using `sybcluster`.

- To manage temporary databases, use the SAP Control Center or follow instructions described in the chapters related to temporary databases.
- To manage logical clusters, use the SAP Control Center or the `sp_cluster logical` and `sp_cluster profile` stored procedures.

For complete `sybcluster` syntax and usage information, see the *SAP ASE Utility Guide*.

Related Information

Temporary Databases [page 156]
Managing the Workload [page 98]

2.1 sybcluster and the SCC Agent

`sybcluster` uses the SCC agent to provide remote management capabilities, in the form of runtime services for managing distributed resources.

SCC agent offers a set of common services and enables processes, such as `sybcluster` and the SAP Control Center, to plug in to an agent to manage server resources and perform various operations.

For more information about the SCC agent, see the *SAP Control Center Installation Guide*.

2.1.1 Starting the SCC Agent for a Custer

(SAP ASE versions 15.7 ESD #1 and later) The SCC agent enables the distributed management of the cluster.

Context

> **Note**
>
> Start the SCC agent on each node that will host instances in the cluster.

Procedure

1. From the `$SYBASE` directory of each instance, source the `SYBASE.sh` file.
2. Start the SCC agent:
   `$SYBASE/SCC-3_3/bin/scc.sh --start --port rmi=<port-number>`

   where `<portNumber>` is any available port. The default port number is 9999; however, this may not be the correct port number for your configuration.

2.2 Starting sybcluster

Start `sybcluster` from the command line.

- The simplest way is to enter:

  ```
  sybcluster -U uafadmin -P
  ```

  or

  ```
  sybcluster -U sccadmin -P
  ```

  "uafadmin" or "sccadmin" are the default user names, and the default password is null or blank. You can also use authenticated SAP ASE database user names and operating system user names.
- You can also start `sybcluster` and, at the same time, specify a cluster, identify a default instance, and connect to the agents on one or more nodes in the cluster. For example:

  ```
  sybcluster -U uafadmin -P -C mycluster -I ase1
  -F "blade1:9999,blade2:9999,blade3:9999"
  ```

  In this example, the `-F` option identifies the node and listening port for each agent in the cluster. If you use this command often, you can create a simple alias for it, such as:

  ```
  sybcluster_mycluster
  ```

Related Information

Authenticating the User [page 31]

2.3 Using sybcluster

Some `sybcluster` commands are available before you connect to a cluster; others are available only after you connect to a cluster.

The `sybcluster` command prompt includes the current cluster and the default instance when these values have been set.

Prompt	When sybcluster is:
>	Not connected to a cluster.
<cluster_name>>	Connected to a cluster.
<cluster_name> <instance_name>>	Connected to a cluster and a default instance has been set.

2.3.1 Commands Active Before Connecting to a Cluster

These commands are active before you connect to a cluster, and are not available after you connect to a cluster.

Command	Description
connect	Connects to an existing cluster.
create cluster	Creates a new cluster.
deploy plugin	Deploys the configuration information for a single instance of the cluster to the SCC Agent.
exit	Exits `sybcluster`.
help	Lists the currently available `sybcluster` interactive commands.
quit	Exits `sybcluster`.
show agents	Displays information about available agents.
upgrade server	Updates SAP ASE to SAP ASE Cluster Edition.

2.3.2 Commands Active After Connecting to a Cluster

These commands are active only after you connect to a cluster:

Command	Description
add backupserver	Configures one or more new Backup Servers on nodes in the cluster not currently configured for Backup Server.
add instance	Adds one new instance to the cluster.
create backupserver	Create Backup Server.
create xpserver	Creates an XP Server.
diagnose { cluster | instance }	Performs a set of checks to ensure the cluster or instance is working properly.
disconnect	Loses all connections to the current cluster and returns the cluster to the unconnected state.
drop backupserver	Drops the Backup Server.
drop cluster	Removes each instance from the cluster and deletes the cluster definition from the cluster configuration file.
drop xpserver	Drops the XP Server.
drop instance	Removes an instance from the cluster.
exit	Exits `sybcluster`.
help	Lists the currently available `sybcluster` interactive commands.
localize	Displays current values for default language, charset, and sort order. Allows you to change the default values and add or remove languages.
quit	Exits `sybcluster`.
set backupserver	Changes the listening port number for Backup Server on one or more nodes.
set cluster	Sets properties for the cluster.
set instance	Sets properties for the instance.
set xpserver	Changes the listening port number for XP Server on one or more nodes.
show backupserver config	Displays the names of the nodes on which Backup Server is configured and the associated listening port number.
show cluster	Displays configuration, log, and status values for the cluster.
show instance	Displays information about an instance.
show membership mode	Displays the cluster's current membership mode, which specifies whether or not the cluster supports Veritas Cluster Server integration.
show session	Displays current agent and discovery information.
show xpserver config	Displays the names of the instances and nodes on which XP Server is configured and the associated listening port number.

Command	Description
shutdown cluster	Shuts down the cluster by executing a Transact-SQL `shutdown` command for each instance in the cluster.
shutdown instance	Shuts down the instance by executing a Transact-SQL `shutdown` command.
start cluster	Starts all instances in the cluster.
start instance	Starts an instance in the cluster.
use	Sets the default instance.

2.4 Connecting to the Cluster

Connect to a cluster before starting it.

- You can connect to a cluster by specifying the required information when you start `sybcluster`.
- You can also start `sybcluster` and connect to the cluster at a later time using the `connect` command.

When you install and configure the cluster, you also install and configure an agent on each node in the cluster. When you start `sybcluster`, provide an authenticated user name and password, and identify the agents on one or more of the cluster's nodes using a direct connect, or discovery method. `sybcluster` can then perform command, control, and discovery operations.

2.4.1 Authenticating the User

Connect to the cluster using any agent-authenticated user name and password.

By default, the user name is "uafadmin" or "sccadmin", and the default password is a null or blank string. For example:

```
sybcluster -U uafadmin -P -C mycluster
```

or

```
sybcluster -U sccadmin -P -C mycluster
```

2.4.2 Identifying the Agents

Identify the agents using a direct connect or discovery method, either when you start `sybcluster` or later, using the `connect to` interactive command.

> **Note**
>
> If the cluster is running when you connect to it, you normally need to identify only one agent on one node in the cluster. To start the cluster or XP Server, however, you must identify the agent on each node in the cluster. `sybcluster` always prompts you when additional information is required.

If you do not know the agent specifications for a cluster, use the `sybcluster show agents` command to discover available agents and clusters on your subnet.

2.4.2.1 Displaying Information About Available Agents

Use `sybcluster` utility's `show agents` command to identify all the configured agents on your subnet, or narrow the search to show information about specific agents.

- To identify all agents, enter:

  ```
  show agents
  ```

  `sybcluster` displays the direct connect address for each agent, its node and cluster name, and other relevant information.
- You can display information about specific agents by restricting discovery or identifying the desired agents. This example displays information about agents on "blade2" of "mycluster":

  ```
  show agents agent "blade2:9999"
  ```

2.4.2.2 Using a Direct Connect Method

You can connect directly to one or more agents by specifying the cluster node and port numbers for the agents responsible for managing the cluster.

Some possible agent specifications are:

- Node name or names of the cluster and optional port numbers – allows you to specify an exact address. If you do not include a port number, `sybcluster` assumes a default value of 9999. This example specifies agents on "blade1", "blade2", and "blade3" of "mycluster":

  ```
  sybcluster -U uafadmin -P -C mycluster -F
  "blade1:1234,blade2:2345,blade3:3456"
  ```

  To specify node and port numbers using `connect` after you have started `sybcluster`, use:

  ```
  connect to mycluster -U uafadmin -P -F "blade1:1234,blade2:2345,blade3:3456"
  ```

- The domain of the node – lets you specify the exact address through the domain name. For example:

```
sybcluster -U uafadmin -P -C mycluster -F "blade1.mydomain.com"
```

2.4.2.3 Using a Discovery Method

`sybcluster` supports two discovery methods that locate the agents and the discovery order.

- User Datagram Protocol (UDP) – is a broadcast discovery method in which `sybcluster` broadcasts a request, and agents located on the same subnet respond. This example uses UDP to look up the location of a node in "mycluster":

  ```
  sybcluster -U uafadmin -P -C mycluster -d "udp()"
  ```

  To perform discovery using `connect` after starting `sybcluster`, enter:

  ```
  connect to mycluster login uafadmin password " " discovery "udp()"
  ```

- LDAP technology – provides lookup capabilities. Each agent registers with the LDAP server, which stores the location of each node and status information.

This example looks up the location of agents for "mycluster" using both discovery methods:

```
sybcluster -U uafadmin -P -C mycluster -d "udp(), ldap(myldapserver:6123)"
```

2.5 Creating a Cluster

Create a cluster using either `sybcluster` or the SAP Control Center. Both methods prompt you for required information and create the cluster for you. You can also create a cluster manually.

For step-by-step instructions for creating a cluster using `sybcluster`, see the SAP ASE Cluster Edition installation guide for your platform and the SAP ASE *Utility Guide*.

For step-by-step instructions for creating a cluster using SAP Control Center, see the *SAP Control Center for SAP ASE*.

2.6 Starting the Cluster

Before starting the cluster, connect to the SCC agent on each node in the target cluster.

Start the cluster:

```
start cluster
```

`sybcluster` prints a cluster description as it starts all instances in the cluster.

Related Information

Connecting to the Cluster [page 31]

2.7 Managing the Cluster

Manage the cluster and its environment.

For complete syntax and usage information for `sybcluster`, see the *SAP ASE Utility Guide*.

2.7.1 Verifying a Cluster

`diagnose cluster` performs a set of checks that ensures that the cluster is working properly.

Enter:

```
diagnose cluster
```

`diagnose cluster` displays cluster information and checks to see if:

- An SCC agent is running on all nodes in the cluster.
- The number of nodes in the cluster does not exceed the maximum instances in the cluster.
- The quorum device exists, and if not, checks to see if the directory has write permission.
- The interfaces file exists for all nodes, and that node names and port numbers do not conflict.
- The primary and secondary protocol specifications do not overlap.
- The Sybase home directories on each node are shared.

2.7.2 Displaying Cluster Information

Use `sybcluster` to display information about the cluster, instances in the cluster, and the cluster environment.

- Display configuration information, primary and secondary protocol values, trace flags used, and the addresses for the quorum device and the master device:

  ```
  show cluster config
  ```

- Display formatted configuration information for the cluster:

  ```
  show cluster config template
  ```

- Display status and heartbeat information for each instance in the cluster:

  ```
  show cluster status
  ```

Status values are:
  - Up
  - Down
  - Undefined
  - Invalid
  - Start
  - Init
  - Quiesce
- Display all log information:

  ```
  show cluster log
  ```

  Limit the output by specifying:
  - The error severity level. For example:

    ```
    show cluster log minseverity 5
    ```
  - A date range for log entries:

    ```
    show cluster log startdate 03:31:08
    enddate 04:30:08
    ```
- Display a specific number of lines from the error log, working backwards from the most recent:

  ```
  show cluster log last 25
  ```
- Display all SCC and JDBC connections to the cluster:

  ```
  show cluster connection
  ```
- Display general information about the cluster and detailed information about agent connections:

  ```
  show session
  ```
- The Symantec Veritas Cluster Server (VCS) manages application services in a cluster environment. If VCS is supported at your site, you can enable VCS on your cluster (see your Veritas documentation). Display the membership mode to find out whether VCS is enabled on your cluster:

  ```
  show membership mode
  ```

  Membership mode values are:
  - `vcs` – the cluster supports VCS integration.
  - `native` – the cluster does not support VCS.

Related Information

Changing the Membership Mode [page 38]

2.7.3 Changing Cluster Configuration Values

Change certain configuration values for the cluster using `set cluster` and other commands. Verify the cluster status by issuing `show cluster status`.

This example changes the maximum number of instances to 4 for "mycluster":

```
set cluster maxinst 4
```

To reset the primary protocol for the cluster to "udp", enter:

```
set cluster primary protocol udp
```

Related Information

Remote Directory Memory Access (RDMA) [page 204]

2.7.3.1 Restrictions for Changing Cluster Configuration Values

You can change certain values when a cluster is down or running.

When the cluster is down, you can change:

- The maximum number of instances
- The active trace flags
- The primary or secondary protocol
- The membership mode

When the cluster is running, you can change:

- The login name or password the agent used to log in to the cluster.
- The default language, character set, and sort order for the cluster.

2.7.3.2 Changing User Names or Passwords

The `sybcluster` utility is a client program that connects to an SCC agent that is configured to run the cluster. When you start `sybcluster`, you provide a user name and password that enables `sybcluster` to log in to the SCC agent.

To change the value of the SCC agent login or password, use the SAP Control Center.

The Unified Agent logs in to the cluster for certain operations, such as when sybcluster issues a `shutdown` command, or when the SCC agent performs cluster heartbeat tasks to determine cluster status. For these

tasks, the SCC agent logs in with sa_role. By default, the SCC agent uses the "sa" login with no password. To change this password, use the `sybclusterset cluster login` command.

For example, to change the password for the "sa" login to "newpassword", enter:

```
set cluster login sa password newpassword
```

The cluster must be running to perform this command.

2.7.3.3 Changing Localization Values

Use the `sybcluster localize` command to view the current values for language, character set, and sort order.

After displaying the current default values, `sybcluster localize` prompts you to either accept each of these values or change them. This example views current values without changing them:

```
localize
```

```
Current default locale properties are:
Default Language - portuguese
Default Charset - mac
Default SortOrder - Binary ordering, for use with the Macintosh character
set(mac).
Options for default Language are:
1. spanish
2. portuguese
3. german
4. us_english
5. thai
6. french
7. japanese
8. chinese
9. korean
10. polish
Enter the number representing the language to be set as defaults: [2]
Options for default charsets are:
1. gb18030
2. eucgb
3. uttf8
Enter the number representing the charset to be set as default: [1]
Options for sort orders are:
1. Binary ordering, for the EUC GB2312-80 character set (eucgb).
Enter the number representing the sort order to be set as default [1]
Do you want to install any language? [Y] n
Do you want to remove any language? [N ]
The cluster mycluster was successfully localized with
default language portuguese, charset gb18030, sortorder
bin_eucgb.
```

To ensure consistency throughout the cluster, shut down and restart the cluster after changing any of the localization values.

2.7.3.4 Changing the Membership Mode

If Veritas Cluster Server (VCS) is supported at your site, you can enable it when you create the cluster. Use the `qrmutil` utility or the Veritas utilities to change the membership mode.

See the *SAP ASE Utility Guide* and your Veritas documentation.

2.7.4 Disconnecting from the Cluster

To close all connections to the current cluster, enter `disconnect`.

2.7.5 Shutting the Cluster Down

You can shut the cluster down gracefully, which allows transactions to complete before SAP ASE Cluster Edition shuts down each instance in the order specified in the cluster configuration file.

```
shutdown cluster
```

To shut down the cluster immediately without waiting for transactions to complete, enter:

```
shutdown cluster nowait
```

> **Note**
>
> If the cluster is running in Veritas Cluster Server (VCS) mode, shut down or start up servers and the cluster using only VCS shutdown and start-up mechanisms. Do not use the `sybcluster shutdown` commands.

2.7.6 Dropping a Cluster

Before you can drop a cluster, make sure that the cluster is in the Down state and the agents are running.

To drop a cluster, use:

```
drop cluster
```

When you perform a drop cluster command:

- SAP ASE Cluster Edition removes cluster and instance entries from the interfaces file.
- Deletes the cluster configuration file.
- Marks the quorum disk as unused.
- Deletes the log file.
- Shuts down the agent.

Confirm each deletion.

2.8 Managing an Instance

Manage instances in the cluster.

2.8.1 Displaying Information About the Instance

Use `show instance` to display configuration, status, and log information about an instance.

- Display configuration information, including the name of the host node, primary and secondary network information, and the path to the log file:

  ```
  show instance <instance_name> config
  ```

- Display status information:

  ```
  show instance <instance_name> status
  ```

  This command displays status information for the named instance:
  - Up
  - Down
  - Undefined
  - Invalid
  - Start
  - Init
  - Quiesce
- Display all log information:

  ```
  show instance <instance_name> log
  ```

  Limit the output by specifying:
  - The error severity level, for example:

    ```
    show instance <instance_name> log minseverity 5
    ```

  - A date range for log entries:

    ```
    show instance <instance_name> log startdate 03:31:08 enddate 04:30:08
    ```

  - A number of lines to display from the error log, working backwards from the most recent:

    ```
    show instance <instance_name> log last 25
    ```

2.8.2 Adding an Instance

You can add an instance to the cluster either by using an input file, or interactively, with `sybcluster` prompting you for required values.

> **Note**
>
> `add instance` creates a local system temporary database for the new instance. Make sure that there is sufficient space on the device.

If you add the instance interactively, SAP ASE Cluster Edition prompts for:

- The instance name, if it was not specified in the command statement
- The name of the node hosting the instance
- The port number of the SCC agent on the node
- The query port number
- The primary and secondary address of the node
- The primary and secondary port specifications

If you add the instance using an input file, make sure the file mirrors the format of the cluster input file (see the SAP ASE Cluster Edition installation guide for your platform), although you need to include definitions only for the new instance. If auxiliary servers are defined, include port and other information required for configuring the auxiliary servers for the new instance.

- To add an instance using an input file, enter:

```
add instance <new_instance> file /$SYBASE/<file_name>
```

- To add an instance interactively, enter:

```
add instance <new_instance>
```

2.8.3 Verifying the Instance

Use `diagnose instance` to perform a set of checks to ensure that the instance is configured properly.

This example verifies the configuration for "ase1":

```
diagnose instance ase1
```

`diagnose instance` displays configuration information for the instance and verifies:

- The query port
- That a JDBC connection is available
- That the instance is available on the public network
- The minimum and maximum port numbers
- The primary and secondary protocol port ranges

2.8.4 Changing the Default Instance

Use the `use` command to set or change the default instance specified in the `sybcluster` command line. After the default instance has been set, you need not specify one in the command line for any interactive command.

This example sets "ase1" as the default instance:

```
use ase1
```

You can override the default instance by including an instance name in an interactive command. However, doing so does not change the default designation.

To remove the default designation, omit the instance name, enter:

```
use
```

2.8.5 Changing Instance Properties

Use `set instance` to change certain properties for the instance.

The instance must be in the Down state to use `set instance`. Verify status by issuing `show cluster status`. This example resets the primary port range from 6123 to 6126:

```
set instance primary port 6123 6126
```

Instance properties you can change are:

- The log path
- Arguments used for starting the instance
- The primary or secondary address of the instance
- The primary or secondary port range used by the instance

2.8.6 Shutting an Instance Down

Use `shutdown instance <instance_name>` command to allow transactions to complete.

This example shuts down "ase1":

```
shutdown instance ase1
```

This example shut down the instance immediately, without waiting for transactions to complete:

```
shutdown instance ase1 nowait
```

If you shut down the last instance in a cluster, the cluster status changes to a down state. If you shut down the instance on the node hosting the cluster coordinator, another node hosts the coordinator.

2.8.7 Dropping an Instance

Use the `drop instance` sybcluster command to drop an instance.

> **Note**
>
> You cannot use `drop instance` to drop the last instance in a cluster. Use `drop cluster` instead.

Before you can drop an instance, perform these steps, if applicable:

1. Log in to instance that you are about to drop.
2. Drop any local temporary user databases. Make sure to perform necessary backup steps first.
3. Drop private devices using `sp_dropdevice`. Make sure to perform necessary backup steps first.
4. Make sure that another instance is up and that the cluster is in the up state.
5. Shut down the instance.

After performing these steps, use sybcluster to connect to the cluster and issue:

```
drop instance <instance_name>
```

SAP ASE deletes entries for the instance from the interfaces file and the quorum device, and notifies the cluster of the topology change. Confirm each deletion.

2.9 Enabling sybcluster After Manually Creating the Cluster

Typically, a cluster is created using `sybcluster` or the SAP Control Center, and SAP ASE Cluster Edition automatically adds configuration information that allows `sybcluster` or the SAP Control Center to connect to the agent on each node.

Prerequisites

If you configure the cluster manually (as described in the SAP ASE Cluster Edition installation guide for your platform), add that configuration information to the agents so that you can use `sybcluster` or the SAP Control Center to manage the cluster.

Procedure

1. If you have not already done so, start the agents on the cluster. See the SAP ASE Cluster Edition installation guide for your platform.

2. Deploy the plug-in. This example deploys the cluster agent plug-in information on the default cluster "mycluster":

```
deploy plugin agent "blade1,blade2,blade3"
```

3. After the agents are specified, SAP ASE Cluster Edition prompts you for the paths to:
   - The quorum device
   - The environment shell script
   - The ASE home directory

> **Note**
>
> You can use `sybcluster` or SAP Control Center to manage the cluster after deploying the plug-in to a single node in the cluster. However, to start the cluster, you must deploy the plug-in to all nodes in the cluster.
>
> You can also use `deploy plugin` to update the values for an existing plug-in.

Related Information

Identifying the Agents [page 32]

2.10 Creating and Managing Auxiliary Servers

Use `sybcluster` to create, drop, configure port numbers, and display current port numbers for Backup Server and XP Server.

2.10.1 Creating Auxiliary Servers

See the SAP ASE Cluster Edition installation guide for your platform to create Backup Server and XP Server using `sybcluster` or SAP Control Center.

See the *SAP ASE Utility Guide* for `sybcluster` syntax and usage information.

If you create multiple Backup Servers, all instances in the cluster require a Backup Server. You can create a single Backup Server for one or more nodes.

If you create XP Servers, you must create them for all instances in the cluster.

2.10.2 Dropping Auxiliary Servers

Use the `drop backupserver`, `drop monitorserver`, or `drop xpserver` to remove an auxiliary server from the cluster. The SAP ASE Cluster Edition prompts for confirmation before dropping a server.

If you configured multiple Backup Servers for the cluster, you will drop all Backup Servers.

If you configured single Backup Servers, you can drop one or all of them. However, `drop monitorserver` and `drop xpserver` drop all XP Servers from the cluster.

This example drops all XP Servers from "mycluster":

```
drop xpserver
```

```
Are you sure you want to drop the XP Servers from cluster "mycluster"? (Y or
N): [N] y
The XP Servers have been dropped for all instances.
```

This example drops the Backup Server from "blade2" of "mycluster":

```
drop backupserver
```

```
Do you want to drop the Backup Server from:
    1. Selected nodes
    2. Cluster
Enter choice: 1
Do you want to drop Backup Server from node "blade1"? [N] n
Do you want to drop Backup Server from node "blade2"? [N] y
Do you want to drop Backup Server from node "blade3"? [N] n
The Backup Server has been dropped.
```

2.10.3 Displaying Listening Port Information

To display current listing port numbers, use show `backupserver config` or `show xpserver config` command.

This example displays Backup Server listening port information for "mycluster":

```
show backupserver config
```

```
Backup Server is configured on the following nodes:
    1. blade1:5001
    3. blade3: 5003
```

2.10.4 Changing Listening Port Information

To change a listening port for an auxiliary server, use `set backupserver` or `set xpserver config`.

This example changes the listening port number for Backup Server for instance "ase3" on "blade3":

```
set backupserver
```

```
Enter the Backup Server port number for instance "blade1": [6011] <CR>
Enter the Backup Server port number for instance "blade2": [6012] <CR>
Enter the Backup Server port number for instance "blade3": [6013] 6666
```

2.11 Upgrading the Server

See the SAP ASE Cluster Edition installation guide for your platform for instructions for upgrading SAP ASE to the current version of the SAP ASE Cluster Edition.

3 Administer Custers with SAP Control Center for ASE

8.10.2

SAP Control Center for ASE allows you to perform administrative tasks for the cluster from the web-interface.

Use SAP Control Center to:

- Create clusters, manage load profiles, add instances to a cluster, display properties of clusters, and manage routes
- Monitor SAP ASE cluster configurations. Monitored resources include memory usage, device I/O, engine CPU utilization, connection information, interprocess communication, workload management, load profiles, routes, and so on.

To manage clusters, you must have connect to permissions granted in the master database.

See the *SAP Control Center for ASE> Manage and Monitor > Manage and Monitor the SAP ASE Environment > Clusters*.

Alternatively, you can perform these tasks using `sybcluster` utility. See Administer Clusters with sybcluster [page 27].

4 Use SAP ASE Cluster Edition with the Veritas Cluster Server

The integration of the SAP ASE Cluster Edition with the Veritas Storage Foundation for Sybase ASE CE (SF for Sybase ASE CE) allows the SAP ASE Cluster Edition to leverage Veritas Storage Foundation storage management technologies, Veritas Cluster Server (VCS) application, and cluster management features. 8.10.2

> **Note**
>
> Integrating the SAP ASE Cluster Edition with the SF for Sybase ASE CE provides cluster availability and data integrity. Do not use other versions of the Veritas Storage Foundation, as they do not contain the necessary integration components.

Features

4.1 Features

The SAP ASE Cluster Edition with SF for Sybase ASE CE include several features.

- Cluster File System – a generic clustered file system that you can use with the SAP ASE Cluster Edition installation files (the files and directories located in `$SYBASE`), database devices, quorum devices, and other application files.
- Cluster Volume Manager – creates the logical volumes that the cluster nodes share.
- Dynamic multipathing – improves storage availability and performance.
- Service group-based application management – provides monitoring and failover capabilities, and allows you to create dependencies between applications so that, if a component fails that the application requires (such as a disk volume), you can fail over the entire application.
- Integrated management of multiple clusters, applications, and database servers using VCS agents and management consoles.
- Support for hardware replication technologies and block-level replication using Veritas Volume Replicator.

8.10.2

4.2 Tasks

The SAP ASE Cluster Edition with SF for Sybase ASE CE perform these key tasks:

- Membership management – the SAP ASE Cluster Edition and SF for Sybase ASE CE maintain their own membership managers. The membership manager has the following responsibilities:
  - Coordinates logging in and logging off the cluster

- Detects failures
- Determines which members of the cluster remain alive in the event of a communications loss (known as arbitration)
- Maintains a consistent cluster view

The Veritas Cluster Membership plug-in (VCMP) allows the SAP ASE Cluster Edition membership service to synchronize with the underlying Veritas membership manager, which avoids a situation where two membership managers are not coordinated, arbitrate a failure differently, and cause the cluster to shut down. Using the VCMP ensures that the SAP ASE Cluster Edition arbitrates in favor of instances that run on nodes within the Veritas membership view.

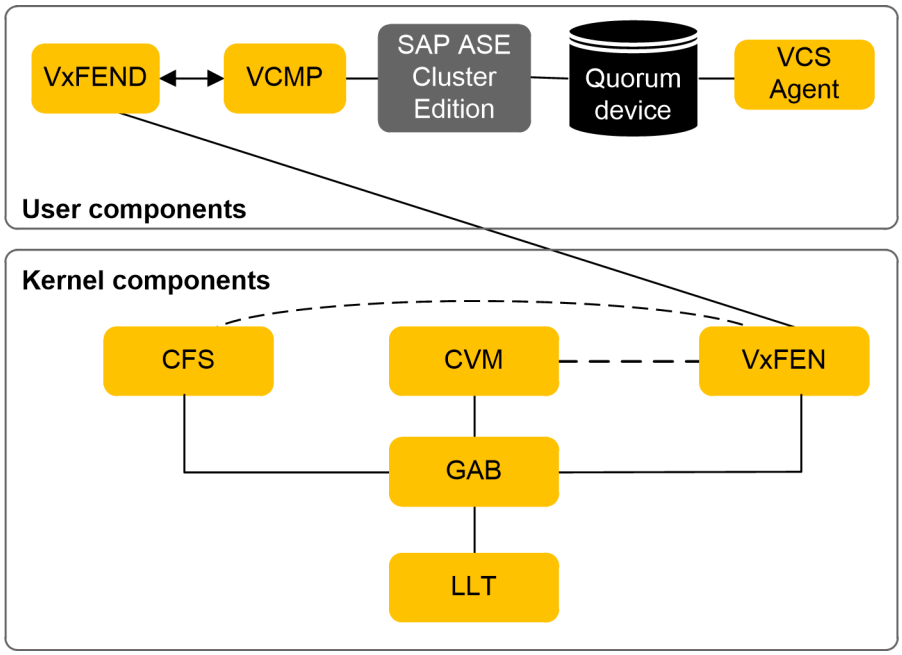
- I/O fencing – The SAP ASE Cluster Edition and the SF for Sybase ASE CE coordinate I/O fencing such that:
  - The SF for Sybase ASE CE manages and performs all fencing
  - The SAP ASE Cluster Edition can communicate to the SF for Sybase ASE CE that a fencing action is necessary because of a membership change

4.3 Components

An integrated SAP ASE Cluster Edition and SF for Sybase ASE CE system includes SAP ASE Cluster Edition, quorum device, and VCMP.

- SAP ASE Cluster Edition – relational database server running on the node
- Quorum device – includes configuration information for the cluster and is shared by all cluster members
- Veritas Cluster Membership plug-in (VCMP) – receives membership change messages from VxFEND and communicates to the SAP ASE Cluster Edition's membership service, which blocks membership changes until the VCMP permits it to proceed

The following figure illustrates these components:



SF for Sybase ASE CE supplies the following components:

- LLT (low latency transport) – allows VCS to communicate across the cluster, detecting heartbeats and failures.
- GAB (global atomic broadcast) – coordinates the VCS membership. The LLT provides the GAB with information about failure events (for example, no heartbeat).
- CVM (cluster volume manager) – receives membership information from the GAB and coordinates with VxFEN.
- CFS (cluster file system) – is a file system that can be simultaneously mounted and accessed by multiple nodes in a cluster. CFS uses a distributed lock manager to maintain consistency across nodes. CVM, with a storage area network (SAN), provides the underlying storage.
- VxFEN (kernel-side I/O fencing control) – receives membership information from GAB and performs fencing, when appropriate.
- VxFEND (user space I/O fencing control daemon) – is a daemon running in the user space (as opposed to a running in the kernel) that communicates with VxFEN. During the membership change, VxFEN sends a message to VxFEND indicating a change of membership. VxFEND informs the SAP ASE Cluster Edition about the membership change through VCMP.
- VCS Agent – is a VCS monitoring agent for the SAP ASE Cluster Edition. If necessary, the VCS agent can trigger a host panic if the SAP ASE Cluster Edition fails. The VCS agent is also used by VCS to start and stop the SAP ASE Cluster Edition.

4.4 Supported Platforms

The SAP ASE Cluster Edition with the Veritas Storage Foundation for Sybase ASE CE is supported on multiple platforms.

For information about supported platforms and required release levels, packages, and patches, see the SAP ASE Cluster Edition release bulletin and installation guide for your platform. For a full list of SF for Sybase ASE CE requirements, see the *Veritas Storage Foundation for Sybase ASE CE Release Bulletin* and *Veritas Storage Foundation for Sybase ASE CE Installation and Configuration Guide*, both from Veritas.

Related Information

Membership Modes [page 56]
Installing and Configuring the SAP ASE Cluster Edition on VCS [page 50]

4.5 Requirements and Limitations of Vcs Mode

The SAP ASE Cluster Edition runs in `native` or `vcs` mode. In `native` mode, it does not coordinate with a the cluster membership manager; in `vcs` mode the cluster membership manager cooperates with the Veritas clusterware.

To use the SAP ASE Cluster Edition in `vcs` membership mode:

- Install and configure SF for Sybase ASE CE on the host nodes.
- Ensure that all SAP ASE Cluster Edition storage, including the master device, user database devices, and that the quorum device come from storage managed by SF for Sybase ASE CE. This can be either a CVM volume or a CFS file.
- Configure the SAP ASE Cluster Edition private networks for the same physical networks as the Veritas LLT links.

Using the SAP ASE Cluster Edition in `vcs` membership mode includes these limitations:

- You cannot run more than one instance on a given hardware node, even if the instances are participating in different SAP ASE clusters.
- You cannot have more than four instances or nodes in a cluster.
- Instance failures can cause a panic restart of a host node. If you plan to run unrelated, critical applications (including other nonclustered SAP ASE servers) on the same node, these applications are shut down when the node restarts.

4.6 Installing and Configuring the SAP ASE Cluster Edition on VCS

To install and configure the SAP ASE Cluster Edition to run on the Veritas cluster subsystem, perform steps described in the Veritas documentation.

8.10.2

Procedure

1. **Note**

   Perform steps 1-6 as described in the *Veritas Storage Foundation Sybase ASE CE Installation and Configuration Guide*.

   Install and configure SF for Sybase ASE CE.
2. Configure I/O fencing in Sybase mode.
3. Create a shared disk group, volume, and mount point for SAP ASE binary installation (`$SYBASE`).
4. Create a shared disk group, volume, and mount point (if using CFS) for the SAP ASE quorum device.
5. Create a shared disk group, volume, and mount point (if using CFS) for the SAP ASE database devices.

6. Ensure the "sybase" user (or equivalent) has appropriate ownership and access for the storage in step 5.

> **Note**
>
> Perform steps 7 and 8 as described in this book in sections Installing the SAP ASE Cluster Edition, Creating a new SAP ASE cluster for Storage Foundation integration, Converting an existing cluster to use SF for Sybase ASE CE.

7. Install SAP ASE Cluster Edition.
8. Create the SAP ASE cluster or prepare an existing cluster for use.

> **Note**
>
> Perform step 9 as described in the *Veritas Storage Foundation Sybase ASE CE Installation and Configuration Guide*.

9. Using the SF for Sybase ASE CE installer, create a VCS service group for your SAP ASE cluster. This puts the cluster into `vcs` membership mode.

Next Steps

Once you configure the SAP ASE Cluster Edition in `vcs` membership mode, start the cluster as part of a VCS service group. If you start the SAP ASE Cluster Edition outside a VCS service group while in `vcs` mode, the SAP ASE Cluster Edition hangs and shuts down.

Related Information

Installing the SAP ASE Cluster Edition [page 51]
Creating a New Cluster for Storage Foundation Integration [page 52]
Converting an Existing Cluster to Use SF for Sybase ASE CE [page 53]

4.6.1 Installing the SAP ASE Cluster Edition

Perform the steps described in the installation guide for your platform to create a cluster using SF for Sybase ASE CE.

You should:

1. Create the release directory (`$SYBASE`) on a storage foundation cluster file system (SFCFS) mount point. Install the SF for Sybase ASE CE prior to installing SAP ASE Cluster Edition. See the *Veritas Storage Foundation for Sybase ASE CE Installation and Configuration Guide*.
2. Before you create the VCS service group, install the SAP ASE Cluster Edition.
3. Create the SAP ASE Cluster Edition in a shared installation instead of a private installation.

4.6.2 Creating a New Cluster for Storage Foundation Integration

Both `sybcluster` and the SAP Control Center automatically detect if the SF for Sybase ASE CE is installed on the host node, and guide you to select the appropriate storage and network interconnects.

When you create a cluster:

- Make sure that all storage is on SF for Sybase ASE CE managed devices, using CVM controlled volumes or SFCFS file systems, with these restrictions:
  - If you access CVM-controlled volumes directly, access them as raw (character) devices, using `/dev/vx/rdsk` (instead of `/dev/vx/dsk/`).
  - Place private devices added specifically for local temporary databases on local, non-SF for Sybase ASE CE storage devices.
  - Do not use storage provided by non-Sybase ASE CE versions of Veritas Storage Foundation. Do not use storage provided by SF for Sybase ASE CE unless you configure the SAP ASE Cluster Edition for `vcs` membership mode. SAP ASE cannot perform VxFEN-controlled I/O fencing in `native` mode, as database corruption may result.
- Configure the VCMP socket path. By default, the VCMP uses `/tmp/vcmp_socket` to communicate with Veritas:
  1. Add the following to the runserver file located in `$SYBASEASE-<version_number>/install/RUN_<instance name>`:

     ```
     export VCMP_SOCKET=<new_vcmp_socket_path>
     ```

     For example:

     ```
     export VCMP_SOCKET=/tmp/my_socket
     ```

  2. Modify the `vxfend` resource:

     ```
     hares -modify vxfend Arguments "%-m sybase -k <new_vcmp_socket_path>"
     ```

     For example:

     ```
     hares -modify vxfend Arguments "%-m sybase -k /tmp/my_socket"
     ```

- The SAP ASE Cluster Edition primary and secondary cluster interconnects must run over networks managed by Veritas LLT. Determine which networks LLT uses from the `/etc/llttab` file. In this example, LLT runs over the `eth0` and `eth1` networks:

  ```
  [admin@sdcc2 ~]$ cat /etc/llttab
  set-node sdcc2
  set-cluster 53
  link eth0 eth-00:19:b9:b0:73:13 - ether - -
  link eth1 eth-00:19:b9:b0:73:15 - ether - -
  ```

  > **Note**
  >
  > The SAP ASE Cluster Edition interconnects require that you configure IP addresses for the links. Because LLT does not require IP addresses, you may need to configure the interconnects separately. See your operating system documentation.

- Use the `ifconfig` command to determine the IP address assigned to a given network. Use this address to configure either the primary or secondary cluster interconnect. In this example, the interface `eth0` has an IPv4 address of 10.22.104.141:

```
[admin@sdcc2 ~]$ /sbin/ifconfig -a eth0
eth0   Link encap:Ethernet  HWaddr  00:19:B9:B0:73:13
       inet addr:10.22.104.141  Bcast:10.22.104.255  Mask:255.255.255.0
       inet6 addr: fd77:55d:59d9:168:219:b9ff:feb0:7313/64 Scope:Global
       inet6 addr: fe80::219:b9ff:feb0:7313/64 Scope:Link
       UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
       RX packets:19710010 errors:0 dropped:0 overruns:0 frame:0
       TX packets:11707065 errors:0 dropped:0 overruns:0 carrier:0
       collisions:0 txqueuelen:1000
       RX bytes:7506123489 (6.9 GiB)  TX bytes:1839949672 (1.7 GiB)
       Interrupt:169 Memory:f8000000-f8012100
```

- Reset the cluster to `vcs` mode after you create the cluster.

4.6.3 Converting an Existing Cluster to Use SF for Sybase ASE CE

If you convert the SAP ASE Cluster Edition to work with the SF for Sybase ASE CE, the Storage Foundation software must manage all database storage, and the SAP ASE Cluster Edition interconnects must operate over LLT links.

4.6.3.1 Relocating Database Storage

Configuration and the size and number of the database devices involved determine the steps to move existing databases from non-Storage Foundation storage to Storage Foundation storage.

4.6.3.2 Reconfiguring Interconnects

Reconfigure interconnects using `qrmutil`.

Procedure

1. Use the `ifconfig` command to determine the proper IP address for the LLT links.
2. Use the `shutdown cluster` command to shut down the cluster.
3. Use `qrmutil` to reconfigure the interconnects. Run this command for each instance in the cluster, using the appropriate instance name, and primary and secondary addresses:

```
qrmutil --quorum_dev=<path_to_quorum_device> --instance=<instance_name>
--primary-address=<address_of_primary_interconnect>
```

```
--secondary-address=<address_of_secondary_interconnect>
```

This example configures the "ase1" instance to use a primary address of 192.168.0.1 and a secondary address of 192.168.0.2:

```
qrmutil --quorum_dev=/sybase_cfs/quorum.dat --instance=ase1
--primary-address=192.168.0.1 --secondary-address=192.168.0.2
```

As an alternative to running this step for each instance, extract the current configuration to a file (as in the next step), edit the file for each instance, and reload the file onto the quorum device using the `--load-config` parameter.

4. Use `qrmutil --extract-config` parameter to create a backup of the new cluster configuration. This example backs up the cluster configuration to the `quorum.bak` file:

```
qrmutil --quorum_dev=/sybase_cfs/quorum.dat
--extract-config=/sybase_cfs/quorum.bak
```

5. Restart the SAP ASE Cluster Edition.

4.7 Managing a Cluster Under VCS Control

To activate software that allows the SAP ASE Cluster Edition and SF for Sybase ASE CE to work together, place the SAP ASE Cluster Edition under VCS control by changing the SAP ASE Cluster Edition membership mode to `vcs`.

To create a VCS service group, see the *Veritas Storage Foundation Sybase ASE CE Installation and Configuration Guide*.

4.7.1 Starting and Stopping Instances

Start and stop an instance that is running in `vcs` membership mode by using a corresponding VCS service group that uses a VCS agent to start and stop the instance. Control the VCS agent with the VCS Cluster Manager (Java console) or the `hagrp` command line utility.

Use the VCS agent to start and stop instances even though the SAP ASE Cluster Edition recognizes the Transact-SQL `shutdown cluster` and `shutdown cluster with nowait` commands, the `sybcluster` utility's `shutdown` command, and the SAP Control Center start and stop selections while in `vcs` membership mode.

See the *Veritas Storage Foundation Sybase ASE CE Administrator's Guide*. For more information about the VCS agent and resource management, see the *Veritas Cluster Server User's Guide*.

4.7.2 When to Add and Remove Instances

Before you can add or remove an instance, add the host node to VCS first, or remove the host node from VCS (optional if removing an instance).

You can perform the tasks in either order, but you cannot start the instance until both tasks are complete.

For information about adding a node to a VCS cluster, see the *SF Sybase CE Administrator's Guide*.

Related Information

Administer Clusters with sybcluster [page 27]

4.7.3 Increasing the Number of User Connections

When you increase the number of user connections, also increase the file descriptor limit.

Veritas running over VCS, however, does not propagate file descriptor limit changes from one user session to another. For example, if you change the file descriptor limits in your current session, subsequent UNIX sessions have the default setting (1024) for the number of file descriptors:

```
ulimit -n 8194
```

Consequently, while the Veritas VCS manages the cluster and brings instances online, it logs into the Linux operating system as the user "sybase" (starting a new UNIX session), and uses the default value for the number of file descriptors.

To ensure that all "sybase" user sessions have the correct file descriptor limit, add the following lines to `/etc/security/limits.conf`:

```
sybase hard nofile 8096
sybase soft nofile 8096
```

4.7.4 Changing the Character Set or Sort Order

Use the following steps to change the character set or sort order.

Procedure

1. Shut down SAP ASE using VCS commands.

2. Change the membership mode to native:

```
qrmutil -Q <quorum_file> --membership-mode="native"
```

3. Restart a single instance in the cluster manually or with `sybcluster`.
4. Perform one of the following steps:
   - Change the character set by running `charset`. For example:

     ```
     $SYBASE/ASE-16_0/bin/charset -Usa -P nocase.srt utf8
     ```

   - Change the sort order by resetting the default `sortorder id` configuration parameter. For example:

     ```
     isql -Usa -P
     sp_configure 'default sortorder id', 101, 'utf8'
     ```

5. Shut down the cluster from `isql`.
6. Restart an instance. SAP ASE reconfigures the character set and sort order, and shuts down.
7. Change the membership mode back to VCS:

   ```
   qrmutil -Q <quorum_file> --membership-mode='vcs'
   ```

8. Restart the cluster using VCS commands.

4.8 Membership Modes

The SAP ASE Cluster Edition includes two membership modes, `native` and `vcs`, which dictate whether the internal membership manager (CMS) for the SAP ASE Cluster Edition functions autonomously (in `native` mode, the default setting) or in cooperation with Veritas clusterware (in `vcs` mode).

4.8.1 Determining the Membership Mode

Determine the current membership mode using the "@@membershipmode" global variable and `sybcluster show membership mode`.

"@@membershipmode" and `sybcluster` return:

- “vcs” if the SAP ASE Cluster Edition is running with the VCS cluster subsystem
- “native” if it is running without the VCS cluster subsystem

:For example:

```
select @@membershipmode
vcs
```

4.8.2 Changing Membership Mode

Generally, you need not change the membership mode because creating a VCS service group automatically changes an instance to `vcs` mode. To change the membership mode, use the `qrmutil --membership-mode` option.

Shut down the clusterusing the `qrmutil` utility:

```
qrmutil --[-Q | quorum-dev=<quorum_device> --membership-mode=<mode>
```

where:

- `vcs` – indicates the SAP ASE Cluster Edition is running with the VCS cluster subsystem
- `native` – indicates the SAP ASE Cluster Edition is running without the VCS cluster subsystem

For example:

```
qrmutil --quorum-dev=/dev/vx/rdsk/quorum_dg/quorumvol --membership-mode=vcs
```

To manually configure the SAP ASE Cluster Edition to load VCMP and activate the VCS integration, add `membership mode = vcs` to the cluster configuration file. After you add this line, the SAP ASE Cluster Edition starts as part of the VCS service group. To configure the SAP ASE Cluster Edition without the Veritas cluster subsystem, enter `membership mode = native` to the cluster configuration file.

This example sets the membership mode to `vcs`:

```
[cluster]
name = isles
max instances = 4
primary protocol = udp
master device = /sybase_cfs/data/master.dat
config file = /sybase_cfg/ASE-16_0/ase.cfg
interfaces path = /sybase_cfg/ASE-16_0/ase.cfg
membership mode = vcs
```

The SAP ASE Cluster Edition starts in native mode if you use neither `vcs` or `native` as the parameter for the `membership mode =` line, or if you do not include the line altogether.

Changing the membership mode in the cluster configuration file has no effect after you generate the `master` database. After the `master` database is generated, use `qrmutil` to change the membership mode:

```
qrmutil --quorum-dev= <quorum_path> --membership-mode=[vcs|native]
```

4.9 Understanding Failure Scenarios

The SAP ASE Cluster Edition coordinates its membership changes with VCS and relies on the SF for Sybase ASE CE to perform I/O fencing. For this reason, the SAP ASE Cluster Edition membership manager waits for VCS to reconfigure its membership and perform I/O fencing.

However, during some failures, SAP ASE requires I/O fencing even if the situation does not result in a VCS membership reconfiguration. To acquire the I/O fencing, SAP ASE passes a message through the quorum

device to the VCS agent on the node to be fenced. When the VCS agent receives the message, it panics the node, which triggers a VCS reconfiguration.

The VCS system reacts in the following ways depending on the situation:

- Host node crashes – detected by VCS. VCS reconfigures cluster membership and performs I/O fencing from the failed node. The SAP ASE Cluster Edition receives a new membership from VCS and performs recovery.
- Host node hangs – results in a VCS LLT timeout and a SAP ASE Cluster Edition heartbeat timeout. VCS reconfigures and performs I/O fencing. The SAP ASE Cluster Edition receives the new membership from VCS and performs recovery. The node that is hanging cannot write to the shared storage, and eventually panics.
- Instance fails or hangs – the remaining instances detect a heartbeat failure and perform membership arbitration. The cluster passes a message to the VCS agent on the host node on which the instance failed, which triggers a host node panic. From this point, the VCS acts in the manner described in the “Host node crashes” situation described above.
- Instance `shutdown with nowait` – the SAP ASE Cluster Edition does not trigger a VCS membership change in VCS membership mode.
- Loss of storage access – the SAP ASE Cluster Edition is unaffected if you configured multipathing, and at least one path to storage remains viable. However, if all paths to storage are lost, VCS induces a host panic, which the cluster handles in the same manner described in "Host node crashes" above.
- Loss of interconnect – response depends on the availability of an alternate communication channel. Both SAP ASE and VCS can switch traffic to an alternate channel. If SAP ASE has no viable channels but VCS does, the cluster reacts in the manner described above in "Instance fails or hangs". If VCS has no viable channels, it acts in the manner described above in the "Host node crashes".

4.10 VCS Troubleshooting

The information in the error log helps you determine the reason for a problem, and possible solutions.

The default error log locations vary between different operating systems.

- SAP ASE Cluster Edition – `$SYBASE/ASE-16_0/install/instance_name.log`
- Veritas – `/var/VRTSvcs/log/engine_A.log`
- Linux – `/var/log/messages`

> **Note**
>
> Many problems can be resolved with Veritas commands (for example, `hagrp` and `hares`). See your Veritas documentation for syntax and usage information.

4.10.1 SAP ASE Cluster Edition Fails to Start

A number of problems can cause the SAP ASE Cluster Edition to not start.

- The SAP ASE Cluster Edition creates a resource fault if VCS cannot start the SAP ASE Cluster Edition or detects that it is already running.
  Clear the resource fault using the `hagrp -clear` command. This example clears the resource faults for an VCS group named "asegrp1":

  ```
  hagrp -clear asegrp1
  ```

- VCS uses the `ps` command to locate the `dataserver` process: do not rename the `dataserver` binary. Validate the `<Home>` attribute for the SAP ASE Cluster Edition resource using `hares -display`. `<Home>` must point to `$SYBASE` and the `dataserver` binary must be located in `$SYBASE/ASE-16_0/bin`. If `<Home>` does not point to the correct value, change the value using the `hares -modify` command.
- VCS uses the `RUN_server_name` file to start the instances. If the `RUN_server_name` files do not exist, VCS uses the parameters defined in the `<Home>`, `<Server>`, and `<Quorum_dev>` variables to build the execution string that starts the instance:

  ```
  <HOME>/ASE-16_0/bin/dataserver -s<server_name> -Q<Quorum_dev>
  ```

If you receive a resource fault, verify that:

- There is a separate `RUN_server_name` file for each instance in the cluster.
- The `RUN_server_name` files include the name of the instance as the suffix.
- Each `RUN_server_name` file is executable.
- Each `RUN_server_name` file contains the correct path to the `dataserver` binary.
- The `<Quorum_dev>` attribute value uses the correct path to an existing quorum device. Use `hares -display` to view the `<Quorum_dev>` value. This example assumes an VCS resource named "aseres1":

  ```
  hares -display aseres1 -attribute Quorum_dev
  ```

  Use `hares -modify` to modify the path:

  ```
  hares -modify aseres1 Quorum_dev /sybase/quorum
  ```

- The `<Server>` attributes correspond to the instance names. Use the `hares -display` command to verify the `<Server>` attributes. This example assumes an VCS resource named "aseres1":

  ```
  hares -display aseres1 -attribute Server
  ```

  Modify the value:

  ```
  hares -modify aseres1 Server ase1 -sys hpcblade1
  ```

- You have enabled the VCS resource by using `hares -display`:

  ```
  hares -display aseres1 -attribute Enabled
  ```

  Use `hares -modify` to modify it if it is necessary:

  ```
  hares -modify aseres1 Enabled 1
  ```

4.10.2 Veritas Log: "Sybase home directory does not exist"

The Sybase home directory (`$SYBASE`) must exist on the CFS mount directory. This means that instances cannot start if the CFS mount directory is not available.

To verify that `$SYBASE` directory is available, validate that the `<ASE>` group is dependent on the `<cfsmount>` group by using `hagrp -dep`:

```
hagrp -dep
```

If it is not currently dependent, reestablish the dependency:

```
hagrp -link asegrp1 cfsmount1 online local firm
```

4.10.3 Instance Log: "Failed to bind to socket"

This error occurs when an instance has a VCMP socket error, The "sybase" user must own the VCMP socket (`/tmp/vcmp_socket`) before it can read and write to the socket and communicate with VxFEN.

Validate that the "sybase" user owns `/tmp/vcmp_socket`:

```
ls -l /tmp/vcmp_socket
```

If the "sybase" user does not own `/tmp/vcmp_socket`, change the ownership:

```
chown sybase /tmp/vcmp_socket
```

4.10.4 Instance Log: "Membership service failed to receive initial message from Veritas cluster membership after 301 seconds. Exiting..."

This error occurs because an instance has a VCMP timeout.

There are a few possible causes:

- The VxFEND resource is unavailable when you bring the `<ASE>` group online. Verify VxFEND resource availability:
  1. Take the VxFEND resource offline:

     ```
     hares -offline vxfend -sys <server name>
     ```

  2. Bring the `ASE` group online:

     ```
     hagrp -online ASE -sys <server name>
     ```

- Another instance is using the socket. Validate that no other `dataserver` processes are running on the node:

  ```
  ps -ef | grep dataserver
  ```

- The VxFEND resource is not enabled. Validate that VxFEND is enabled:

  ```
  hares -display vxfend -attribute Enabled
  ```

  If it is not enabled, enable it:

  ```
  hares -modify vxfend Enabled 1
  ```

- The SAP ASE resource is not dependent on the VxFEND resource. Validate the `ASE` resource dependency:

  ```
  hares -dep
  ```

  If it is not dependent, reestablish the dependency using `hares -link`:

  ```
  hares -link aseres1 vxfend
  ```

- Another SAP ASE or VxFEND resource conflicts with a resource startup. Validate that no other SAP ASE or VxFEND resource is enabled:

  ```
  hares -display
  ```

  Rectify any problems:

  ```
  hares -modify ASE2 Enabled 0
  ```

4.10.5 Instance Log: "Failed to open quorum device 'device_path'. OS error 13, 'Permission denied'"

This error occurs because permissions on the VCS raw volumes do not include access for the "sybase" user.

Grant the "sybase" user read and write permission to the VCS volumes:

```
vxedit -g sybasedg set user=sybase group=sybase mode=660 quorum
vxedit -g sybasedg set user=sybase group=sybase mode=660 master
vxedit -g sybasedg set user=sybase group=sybase mode=660 sybasehome
```

4.10.6 Instance Log: "basis_dsizecheck: attempt to open device 'device_path' failed, system error is: Permission denied"

This error occurs for the same reasons as another instance log error: "Failed to open quorum device 'device_path'. OS error 13, 'Permission denied'". Rectify with the same steps.

4.10.7 Instance Log: "The configuration area in master device appears to be corrupt."

This error occurs when the `<master>` database is corrupted or the clustered file system causes hangs. This problem may occur because the disk array is not properly configured for Veritas.

You may also see these messages in the Linux log:

```
Linux log: READ CAPACITY failed
Linux log: reservation conflict
Linux log: attempt to access beyond end of device
```

To rectify:

- Verify that the disk array firmware is up to date.
- (Linux only) Verify that the disk initialization format in the Veritas Volume Manager is sliced.
- Correct the disk array connection mode configuration settings so that the logical unit numbers (LUNs) reserved for one host are visible to the other host, and verify that the SCSI-3 fencing is properly enabled on the array. Generally, these settings are:
  - Set Host Mode Option = 19 (ON)
  - Set Host Mode Option = 254 (ON)
  - Set Host Mode Option = 186 (ON)
  - Set Host Mode Option = 60 (ON)
  - Host Mode = 0A
  - Path Switch Mode (Active/Active) = 1 (Enabled)
  - No_RSV_Conf Mode = 1 (Enabled)
  - Persistent RSV Cluster Mode = 1 (Enabled)
  - Unique Reserve Mode = 1 (Enabled)

See your Veritas documentation for settings specific to your operating system and hardware.

4.10.8 Veritas Log: "Path not found"

This error occurs because VCS cannot locate the UNIX `sh` command to start the `dataserver` binary. Veritas truncates the start of the `<$PATH>` environment variable, so verify that Veritas is locating the `sh` command. Check the value for `<$PATH>` and restart Veritas.

Procedure

1. Check the value for `<$PATH>`:

   ```
   echo <$PATH>
   ```

2. Change the value of `<$PATH>` to include the `sh` command:

   ```
   export PATH=/bin:/sbin:/usr/bin:/usr/sbin
   ```

3. Stop Veritas:

   ```
   Issue hastop -local
   ```

4. Restart Veritas:

   ```
   hastart
   ```

4.10.9 VCS Shuts Down the Instance After It Starts and Issues Resource Faults

If VCS shuts down the instance after it starts and issues resource faults, clear the resource faults with `hagrp -clear`.

For example, this command clears the resource faults for an VCS group named "asegrp1":

```
hagrp -clear asegrp1
```

Check the error log. This error may result in the instance writing a number of errors to the error log:

- `Failed to identify instance` – VCS is using the wrong name to start the instance. To rectify:
  - Verify that the each `RUN_server_name` file includes the correct path to the quorum device, and that the value for `server_name` is the same as the name of the instance.
  - Use `qrmutil` to validate the quorum configuration, specifically the instance and host names:

    ```
    qrmutil -Q<quorum_path> --display=all
    ```

- `Shutdown with nowait detected` – VCS cannot determine if the instance is running. VCS uses the UNIX `ps` command with the server name in the execution string to verify whether an instance is running. To make sure the `ps` command can find each instance:

- Use `hares -display` to verify that the `<Server>` attributes for the `ASE` resource correspond to the instance names. This example displays the instance names for the "aseres1" resource:

  ```
  hares -display aseres1 -attribute Server
  ```

  If necessary, change the values:

  ```
  hares -modify aseres1 Server ase1 -sys hpcblade1
  ```

  If an instance requires a long time to start, Veritas may stop the `dataserver` process during startup and issue a resource fault. To prevent this, allow instances more time for startup by adjusting the following Sybase VCS monitoring parameters:
  - Adjust the time between monitor checks for the `ASE` resource. For example:

    ```
    hatype -modify Sybase MonitorInterval 60
    ```
  - Adjust the number of monitor checks to confirm the `ASE` resource is online. For example:

    ```
    hatype -modify Sybase OnlineWaitLimit 10
    ```
  - Adjust the number of failed monitor checks before issuing a resource fault for the `Sybase` resource. For example:

    ```
    hatype -modify Sybase ToleranceLimit 2
    ```

- `Engine exited with signal 11` – (Linux only) the instance fails. To rectify, reset the `<LD_POINTER_GUARD>` environment variable in the `RUN_server_name` files:
  - On SUSE:

    ```
    export LD_POINTER_GUARD=1
    ```
  - On RedHat:

    ```
    export LD_POINTER_GUARD=0
    ```

4.10.10 VCS Cannot Shut Down the Instance Resource

This problem occurs when VSC has trouble using `isql` to connect to an instance. VCS uses `isql` to connect and shut down instances. `isql` uses the interfaces file for the connection parameters.

Validate that:

- The interfaces file is defined in the release directory (`$SYBASE`) and is configured correctly.
- The interfaces file contains definitions for each defined instance.
- The release directory contains a `SYBASE.sh` or `SYBASE.csh` file and is configured correctly.

VCS uses the `SA` and `SApswd` commands with `isql` to connect and shut down instances. Verify the `SA` and `SApswd` attributes for the `<ASE>` resource:

```
hares -display aseres1 -attribute SA
hares -display aseres1 -attribute SApswd
```

If necessary, change the attributes. Use `hares -modify`:

```
hares -modify aseres1 SA sa
hares -modify aseres1 SApswd ""
```

4.10.11 Finding Resource Faults for VCS Groups

Find resource faults from the `cfsmount` group.

- Verify that the `cfsmount` group is dependent on the `cvm` and `cvmvoldg` group:

  ```
  hagrp -dep
  ```

  Fix any dependencies:

  ```
  hagrp -link cfsmount1 cvm online local firm
  hares -link cfsmount1 cvmvoldg1
  ```

- Verify that the VCS mount and volume disk groups are enabled on all nodes:

  ```
  vxdg -o alldgs list
  ```

  Re-enable:

  ```
  vxdg deport <disk_group_name>
  vxdg -s import <disk_group_name>
  ```

- Verify that the VCS mount and volume disks are online:

  ```
  vxdisk list
  hastop -local
  ```

  If necessary, restart VCS on each node:

  ```
  /etc/init.d/vxfen stop
  /etc/init.d/vxfen start
  hastart
  ```

- If CFS mounts fail because VCS volumes are in use, you may need to restart all nodes in the cluster to clear any conflicts. See your operating system documentation.

If you find resource faults from the `cvm_clus` group, verify that this resource is dependent on the `cvm_vxconfigd` resource:

```
hares -dep
```

Fix any dependencies:

```
hares -link cvm_clus cvm_vxconfigd
```

If you find resource faults from the `vxfsckd` group, verify that the this resource group is dependent on the `cvm_clus` resource:

```
hares -dep
```

Fix any dependencies:

```
hares -link vxfsckd cvm_clus
```

4.10.12 VCS Fails to Start

This error occurs when a node cannot join the VCS cluster.

- This message in the error log means that the node cannot join the cluster because VCS cannot start I/O fencing:

  ```
  CVMCluster:???:monitor:node - state: out of cluster
  ```

  1. Verify that the `vxfencoordg` disk group is enabled on all nodes:

     ```
     vxdisk list
     ```

  2. If the `vxfencoordg` disk group is not enabled, deport and re-import the group:

     ```
     vxdg deport vxfencoorddg
     vxdg -t import vxfencoorddg
     ```

  3. Verify that the `vxfencoordg` disks are online:

     ```
     vxdisk list
     ```

  4. If the `vxfencoordg` disks are not online, restart VCS on each node:

     ```
     hastop -local
     /etc/init.d/vxfen stop
     /etc/init.d/vxfen start
     hastart
     ```

  5. Verify reservations exist on the `vxfencoordg` disks:

     ```
     vxfenadm -g
     ```

     If the reservations do not exist, stop the VCS, clear the VCS reservations, and restart VCS:

     ```
     hastop -local
     /etc/init.d/vxfen stop
     vxfenadm -g all -f /etc/vxfentab
     vxfenclearpre
     /etc/init.d/vxfen start
     hastart
     ```

     If a new node joins an existing cluster, you may need to reapply the group membership configuration:

     ```
     gabconfig -x
     ```

- This message in the error log means that the Veritas configuration daemon cannot start:

  ```
  ERROR: IPC Failure: Configuration daemon is not accessible
  ```

  Clear and restart the configuration daemon:

  ```
  vxconfigd -k
  ```

```
"vxiod set 10
"vxconfigd -m disable
"vxdctl init
"vxdctl initdmp
"vxdctl enable
```

- This error in the VCS error log means that the disk group initialization failed when you ran `vxdg init`:

  ```
  Device sda cannot be added to a CDS disk group
  ```

  By default, `vxdg` initializes disks as CDS. Rerun the `vxdg` command with `cds=off`:

  ```
  vxdg init <disk_group> cds=off
  ```

5 Private Installation Mode

When you install and create your cluster, choose whether to configure the cluster in shared or private installation mode:

- Shared installation – supports a shared file system created using a Network File System (NFS) or a cluster file system. A cluster created using a shared installation supports a single `$SYBASE` installation directory, SAP ASE home directory, and server configuration file.
- Private installation – requires a separate `$SYBASE` installation directory, SAP ASE home directory, and server configuration file for each instance. Parity among server configuration files is maintained by the master configuration file on the quorum device.

> **Note**
>
> Use LDAP as your directory service when using a private installation.

If the installation mode is private, the cluster input file automatically contains this entry:

```
installation mode = private
```

If the installation mode is shared, the cluster input file automatically contains this entry:

```
installation mode = shared
```

For instructions about installing and configuring the cluster in private or shared installation mode, see the SAP ASE Cluster Edition installation guide for your platform.

5.1 Maintaining the Server Configuration Files

View the current version number of any configuration file using the `qrmutil` utility.

The syntax is:

```
qrmutil --quorum-dev=/dev/raw/raw101 --ase-config-version
```

A configuration file version number tracks the most recent copy of the server configuration file. This information is stored on the quorum device and in each server configuration file. Every time a new configuration file version is added to the quorum device, the version number is incremented and SAP ASE dumps the new server configuration file and version number to each instance.

For the complete `qrmutil` syntax, see the *SAP ASE Utility Guide*.

5.1.1 Changing Configuration Options at Runtime

While an instance is running, update server configuration options using `sp_configure`. If an option changes, SAP ASE automatically updates the master configuration file, increments the version number, updates the `sysconfigures` table, and instructs all instances to dump a new copy of the configuration file.

When using `sp_configure` with the SAP ASE Cluster Edition:

- Configure an instance-specific value that applies only to the instance to which you are currently connected.
- Configuration options that are "strictly cluster-wide" – that is, the small group of configuration options that are the same on all instances – cannot be made instance-specific.
- Reconfiguring a cluster-wide value does not overwrite an instance-specific configuration.

If an instance is not running during the configuration process, its configuration file and version number are not updated. You can either let SAP ASE detect this when the instance starts, or you can check the version number and dump a new copy from the quorum device:

```
qrmutil --quorum-dev=/dev/raw/raw101 --ase-config-version
qrmutil --quorum-dev=/dev/raw/raw101 --ase-config-extract=mycluster.cfg
```

Related Information

If the Version Numbers Do Not Match at Start-Up [page 70]

5.1.2 Manually Changing the Configuration File

You can manually change the configuration file when the configuring instance is not running.

Procedure

1. Extract the configuration file. For example:

   ```
   qrmutil --quorum/dev/raw/raw101 --ase-config-extract=mycluster.cfg
   ```

2. Edit the configuration file. For example:

   ```
   vi mycluster.cfg
   ```

3. Restart the instance.

Results

When you restart the instance, SAP ASE Cluster Edition automatically increments the version number, updates the `sysconfigures` table with the new version number, writes the new configuration file to the local directory and to the quorum device, and instructs all instances to dump a new copy of the configuration file from the quorum device.

> **Note**
>
> When the configuring instance is not running but the coordinator is running, you can change instance-specific parameters but not cluster-wide parameters.

5.1.3 If the Version Numbers Do Not Match at Start-Up

SAP ASE Cluster Edition checks the file version number when each instance is started. The version number in the server configuration file must match the version number that is stored on the quorum device.

If the file version numbers do not match, SAP ASE prints an error message, extracts the latest copy of the configuration file from the quorum disk to the local file system, and aborts the instance start-up.

At this point, you can either restart the instance using the latest configuration file, or manually edit the file and then restart the instance:

- If the version numbers now match, and there has been no change to the configuration file, you can successfully restart the instance.
- If you edit the configuration file, the version numbers match, but the content of the configuration file has now changed. Restart the instance: SAP ASE automatically increments the version number, updates the `sysconfigures` table with the new version number, writes the new configuration file to the local directory and to the quorum device, and instructs all instances to dump a new copy of the configuration file from the quorum device.

5.1.4 Recommended Configurations

Do not manually change the "config file version" parameter in the server configuration file. This should remain a server-generated number.

Update configuration file parameters only on the start-up instance. If you update configuration file parameters on another instance, when that instance starts up, it generates a new configuration file that removes all user changes from other instances.

6 Client Applications and Client/Server Interaction

You can modify applications that make Open Client/Client-Library calls to support clusters.

> **Note**
>
> DB-Library can connect to an instance in a shared-disk cluster, but it does not support shared-disk or high-availability functionality.

The versions of Open Client, jConnect for JDBC, ODBC, OLE DB, and ADO.NET that ship with the SAP ASE Cluster Edition support login redirection, connection migration, and extended high-availability failover. These features, which use Open Client 15.0 libraries, are enabled automatically:

- Login redirection – the ability of an instance to redirect an incoming client connection to another instance prior to acknowledging the login. Login redirection occurs during the login sequence. The client application does not receive notification that it was redirected.
- Connection migration – occurs when an existing client is transferred from one instance of a cluster to another.
- Extended high-availability failover – in an extended failover configuration, SAP ASE provides a list of failover addresses to clients that are aware of high-availability when they connect. This allows high-availability-aware clients or applications to fail over multiple times if the instance to which they are connected fails.
  Although these clients need not have a HAFAILOVER entry in their interfaces file or directory services, if they do have an HAFAILOVER entry, the clients continue to use this entry until SAP ASE sends them a list of failover addresses or servers to connect to. The clients always use the latest list that SAP ASE provides.

To implement login redirection and connection migration, make sure the application uses a current copy of the client libraries:

- If the application is linked to shared libraries – include the new client libraries in the library search path before the old libraries.
- If the application is statically linked – relink the application.

To implement extended failover, make sure that the application uses a current copy of the client libraries and that you have enabled high availability. For information about enabling high availability, see *SAP ASE Using SAP Failover in a High Availability System*.

Related Information

Connection Migration [page 76]

6.1 Open Client and OCS Version Support

Login redirection, connection migration, and high extended availability failover are supported for the following Open Client and OCS versions:

- Login redirection – Open Client version 15.0
- Extended high availability – OCS version 15.0 ESD #3
- Migration – OCS version 15.0, ESD #8

The SDK component that supports these versions is OpenClient/Client-Library.

6.2 Failover in Client-Library Applications

While any existing application can connect to the SAP ASE Cluster Edition, you may need to change application code to use extended high-availability (HA) capabilities.

- Existing HA applications based on existing SAP ASE HA functionality require no application code changes.
- Existing non-HA applications may benefit from some aspects of the HA capabilities of the SAP ASE Cluster Edition with little to no code changes. However, in these cases, failover is not transparent: the application receives an error message when failover is first detected. The user must resubmit the batch or transaction to initiate failover.

To make failover transparent to users, the application must actively check for failover error status, and automatically resubmit the batch or transaction.

In all cases, update the Client-Library used by the application to use the cluster-related HA capabilities.

6.2.1 Enable Failover for Non-HA Applications

For `isql`, specify the `-Q` option when connecting to SAP ASE. For applications linked with Client-Library, set a corresponding connection property that enables failover.

6.2.2 Enable Failover in Client-Library Applications

Set the CS_FAILOVER property to enable failover in Client-Library applications.

Procedure

1. Set CS_HAFAILOVER property at either the context or the connection level using the `ct_config` or `ct_con_props` Client-Library API calls:

   ```
   ct_config (context, action, CS_HAFAILOVER, buf, buflen, &outlen);
   ct_con_props(connection, action, CS_HAFAILOVER, buf, buflen, &outlen);
   ```

   For more information about the CS_HAFAILOVER property, see the *SAP Open Client Client-Library/C Reference Manual*.
2. If you attempt to connect to an instance that is down, Client-Library tries all the query entries for the instance name in the interfaces file until one of them works, or it has none left to try. Include query lines in the client-side interfaces file for all instances. Applications can connect to the cluster, which is represented by a series of interfaces file query entries.

   For information about the interfaces file, see the installation guide for your platform.
3. When a successful failover occurs, Client-Library issues a return value named CS_RET_HAFAILOVER, which is specific to several Client-Library API calls, including:

   ```
   ret = ct_results(cmd, result_type)
   ret = ct_send(cmd)
   ```

   The API call returns CS_RET_HAFAILOVER during a synchronous connection (a routine that requires a server response and blocks until the response is received). In an asynchronous connection (a routine that requires a server response returns CS_PENDING immediately), these APIs issue CS_PENDING, and the `callback` function returns CS_RET_HAFAILOVER. Depending on the return code, perform the required processing, set up the context, and send the next command to be executed.
4. Rebuild your applications using the Open Client SDK with a version that is at least equal to the version of the Open Client SDK shipped with the SAP ASE Cluster Edition.

   For information about configuring applications for high availability, see *SAP ASE Using SAP Failover in a High Availability System*.

6.3 Extended High-Availability Failover

SAP ASE provides a list of failover addresses to high-availability-aware clients when they connect. This allows high-availability-aware clients or applications to failover multiple times whenever the instance to which it is connected becomes unavailable.

If the instance has not sent a failover list to the client, the client uses the HAFAILOVER entry information in the interfaces file.

This example allows a high-availability-aware client to fail over if there is a network failure during login before the instance sends the extended high-availability list:

```
ase1
        query tcp ether blade1 19786
ase2
        query tcp ether blade2 19786
mycluster
        query tcp ether blade1 19786
        query tcp ether blade2 19786
        hafailover mycluster
```

The HAFAILOVER entry should use the cluster alias as the server name since a client application tries each query line until it establishes a connection to an instance in the cluster.

> **Note**
>
> Extended HA failover requires Open Client 15.0, ESD #3 or later. The client libraries in the SAP ASE Cluster Edition contain ESD #8.

Open Client uses the CS_PROP_EXTENDEDFAILOVER property for extended failover. See the *SAP Open Client Client-Library/C Reference Manual* for more information.

6.3.1 Differences Between HA Failover and Failover in Clusters

From the client side, enabling high availability means that the application receives the error code CS_RET_HAFAILOVER during a network failure, and the Client-Library automatically reconnects to a server.

In high availability, the application again attempts to connect first to the original server, and then tries the secondary server (as set in the interfaces file entry for the primary server) if the connection fails. If you use the Open Client libraries shipped with the SAP ASE Cluster Edition, the instance sends a failover target list to the client. The application uses this list to determine where to connect when the instance fails.

Whether you use high availability or the SAP ASE Cluster Edition, failover can occur at any point.

6.4 Using isql in a Clustered Environment

By default, use `isql` to connect to a server instance in a shared-disk cluster environment. However, to connect to an instance in a shared-disk cluster and turn on high-availability failover or extended high-availability failover for the client, you must start `isql` using the `-Q` option.

> **Note**
>
> Although `isql -Q` can use extended high-availability capabilities, it is not transparent. When an instance fails, `isql` receives an error and you must resubmit the batch or transaction.

See the *SAP ASE Utility Guide*.

6.5 Login Redirection

Login redirection occurs at login time when an instance tells a client to log in to another instance because of load considerations.

Login redirection is enabled automatically when an instance uses the Open Client 15.0 client libraries. You need not perform any additional configuration for client redirection.

The server workload manager uses login redirection to send incoming connections to specific instances based on the logical cluster configuration and the cluster's current workload.

If you attempt to connect to an instance that is down, behavior is the same as with a nonclustered SAP ASE: the client tries all entries in the directory service of a given server until it can connect successfully. Because of this, your server entries in the directory service should contain connection information for all instances in the cluster.

The following example includes the servers "ase1", "ase2", "ase3", and "ase4", on machines "blade1", "blade2", "blade3", and "blade4", running in the cluster "mycluster":

```
ase1
   query tcp ether blade1 19786
ase2
   query tcp ether blade2 19786
ase3
   query tcp ether blade3 19786
ase4
   query tcp ether blade4 19786
```

```
mycluster
   query tcp ether blade1 19786
   query tcp ether blade2 19786
   query tcp ether blade3 19786
   query tcp ether blade4 19786
```

For example, if a client connects to cluster "mycluster", it first tries to connect to the "ase1" instance. If "ase1" is down, it tries "ase2", the next entry in the interfaces file, and so on. After a successful connection, the workload manager may redirect the client to another instance based on workload rules.

Although instances are tried in the order specified in the interfaces file, a connection attempt may take a while to fail when hosts or the network are unreachable or down. Expedite the retry attempt by adding a login timeout to the connection information.

In the example above, if you specified a login timeout interval that is shorter than the default for the connecting client, the client could attempt to connect to instance "ase2" more quickly.

See the `isql -l` parameter description in the *SAP ASE Utility Guide* and CS_LOGIN_TIMEOUT property in the *SAP Open Client Client-Library/C Reference Manual*.

6.5.1 Setting Connection Properties for Login Redirection

Configure login redirection by setting CS_PROP_REDIRECT and CS_DS_RAND_OFFSET connection properties.

- CS_PROP_REDIRECT – enables and disables login redirection.
- CS_DS_RAND_OFFSET – disables or enables making first query entry randomly retrieved from the directory service lookup for `ct_connect`. By default, this property is set to false.

See the *SAP Open Client Client-Library/C Reference Manual*.

6.6 Connection Migration

Connection migration occurs when an existing client is transferred from one instance of a cluster to another.

For example, a connection may migrate for load balancing, or because the instance to which it is currently connected is brought down for maintenance. This transfer is transparent to the client application.

Connection migration allows the cluster to balance the load throughout the logical cluster. It allows the workload manager to gracefully move existing connections between instances during administrative failover, failback, or logical clusters going offline. The workload manager can use migration for dynamic load distribution, during which some existing connections are migrated between instances to more evenly distribute load.

6.6.1 Enabling and Disabling Connection Migration

Connection migration is enabled automatically when an instance uses the Open Client 15.0 client libraries. You need not perform any additional configuration for connection migration.

To enable or disable connection migration, use the CS_PROP_MIGRATABLE connection property. CS_PROP_MIGRATABLE is on by default.

See the *SAP Open Client Client-Library/C Reference Manual*.

6.6.2 Difference Between Migration and Failover

Migration is a planned, controlled event that a server requests. A failover is an unplanned event caused when a server crashes or network disconnects.

Although applications are unaware of – and need no code written for – migration, code applications explicitly for failover support.

Migration restores the full session context on the new instance, but failover does not; it is up to the application to restore its own context.

6.6.3 When Can Migration Occur?

The workload manager can initiate a migration when a request can successfully be sent to a client.

Specifically, migration can occur:

- To connections that have completed their login:
  - After an instance receives a new batch from the client, but before the batch is parsed and executed.
  - After an instance completes the processing of a client batch but before it sends a final completion to the client.
  - When an instance is not executing any batch on behalf of the client.
- According to the workload manager's algorithm. The workload manager may migrate certain clients for load balancing or because the current instance is being brought down for maintenance.
- When a connection's context accommodates migration. The workload manager targets connections for migration, but these connections migrate only when their context allows this. In particular, no migration can occur within a transaction.

> **Note**
>
> Connection migration is unsupported for connections that are executing Java.

Related Information

Using Java in a Clustered Environment [page 90]

6.6.4 Criteria for Migration

Migration is an asynchronous event; a request is issued for the task to migrate, and the task migrates only when it reaches a quiescent state.

For the SAP ASE Cluster Edition, a quiescent state is one that:

- Is not executing a query batch
- Has no open transactions
- Has no session-level temporary tables
- Has no declared cursors
- Has not changed its password since its initial connection
- Has not run `set user` or `set proxy`
- Is not bound to an engine using the logical process manager
- Is not using SAP ASE Active Messaging
- Is not a logical connection associated with an inbound site handler
- Has not kept a database open in single-user mode.

6.6.5 Migrated Contexts

When SAP ASE Clustered Edition migrates an existing connection to another instance, it also migrates some of the context from the existing connection, such as the current database and `set` options.

The source instance propagates the client's full context to the destination instance when it has successfully completed the migration, and the destination instance retrieves the context. The client's full context is restored after the migration is complete, making migration completely transparent to the client. However, a migrated connection acquires a new SPID.

The client's context consists of the following:

- The name of the current database
- Any pending batch of commands if the migration occurs in pre-batch mode
- The client's login record
- The client's language and character sets
- The client's capabilities
- Any monitor counters and statistic
- Any roles
- Any `set` options
- Trace flags 3604 and 3605

6.6.6 Client Migration Configuration Parameters

The `idle migration timer` and `session idle timer` configuration parameters control when idle clients migrate. The sum of these two parameters determines the upper limit for the number of seconds during which migration is expected to complete.

The default setting for `idle migration timer` is 60, measured in seconds. The default setting for `session idle timer` is 600, measured in seconds.

If you set `idle migration timer` to 0, the instance closes the connection on which a migration request was issued immediately after it sends the migration request. If you set `session idle timer` to 0, the instance invalidates any idle migration not completed before `idle migration timer` expires.

Setting `session idle timer` to a high value increases the chance for idle connections to migrate successfully. However, the instance must preserve the contexts of the migrating clients for a long time. It also means that setting both `idle migration timer` and `session idle timer` to 0 disables migration. For more information about these configuration parameters, see the *SAP ASE System Administration Guide: Volume 1 > Setting Configuration Parameters > Configuration Parameters*.

Both parameters set to the default

- If an instance issues a migration request against an idle client that cannot process asynchronous notifications, and the user or the application issues a command on the client before the expiration of the `idle migration timer`, the client migrates immediately and the command is executed on the

destination instance. "Read ahead" clients read any pending data from the network before the application requests it.

- The instance closes the connection to the client if all of the following are true:
  - Both `idle migration timer` and `session idle timer` are set to their default values.
  - An instance issues a migration request against an idle, non-read-ahead client.
  - The client remains idle for the duration of `idle migration timer`.
- If a command is issued on the client during the first 600 seconds (the default value of `idle migration timer`) after an instance connection closes, the client migrates successfully and the command is executed on the target instance.
- An instance closes the connection to the client if all of the following are true:
  - Both `idle migration timer` and `session idle timer` are set to their default values.
  - An instance issues a migration request against an idle, non-read-ahead client.
  - No command is issued against the client before `idle migration timer` expires.
- An instance invalidates the migration if all of the following are true:
  - Both `idle migration timer` and `session idle timer` are set to their default values.
  - An instance issues a migration request against an idle, non read-ahead client.
  - No command is issued against the client before `idle migration timer` expires.
    The client attempts to migrate when it detects the migration request, because a command is issued. However, the instance rejects the migration request and the client tries to continue on the initial instance. Because the connection to the instance is closed (`idle migration timer` timed out), the client attempts an HA failover if it has HA capability – otherwise, it reports a disconnected state to the application.

Idle migration timer set to 0, session idle timer set to 600

If an instance issues a migration request against an idle, non-read-ahead client, the instance closes the connection to the client immediately after sending the migration request. If a command is issued against the client during the initial 600 seconds, the client migrates successfully, otherwise, it fails in the same manner described in the previous bullet.

Idle migration timer set to 60, session idle timer set to 0

The instance issues a migration request against an idle, non-read-ahead client. If a command is issued against the client before `idle migration timer` expires, the client migrates successfully. However, if no command is issued before `idle migration timer` expires, the client can migrate, since `session idle timer` is set to 0.

Idle migration timer and session idle timer set to 0

If both parameters are set to 0, neither instance migrates clients.

6.7 Remote Procedure Calls in a Clustered Environment

Remote procedure calls (RPCs) allow clients to initiate stored procedures on a remote server. The remote server runs the system procedure in its context as if it was requested by a local client, and sends back results to the original server over the network.

In addition to server-to-server RPCs, the SAP ASE Cluster Edition includes three additional classes of RPCs:

- The first class involves an RPC where the remote server is a cluster.
- The second class involves an RPC where the local, or originating, server is a cluster instance.
- The third class involves an RPC where both the local and remote servers are instances in the same cluster.

In the nonclustered SAP ASE, the default value for `cis rpc handling` is 0. In the SAP ASE Cluster Edition, the default value is 1, which forces RPC handling to use Component Integration Service (CIS) as the default RPC handling mechanism instead of the site handler.

A cluster instance identifies itself to remote servers using the name of the cluster, not the name of the instance. The `<@@servername>` global variable returns the name of the cluster.

6.7.1 RPCs Where the Remote Server is a Cluster

The SAP ASE Cluster Edition supports a single-system image, which means that using a cluster as a remote server has minimal impact on the server sending the RPC request.

If the server sending the RPC is using `cis rpc handling`, the cluster perceives the inbound request as a regular client connection. The workload manager attempts to route the RPC to the appropriate instance and logical cluster based on the configured routing rules. The workload manager rules may dictate login redirection, as long as the initiating server indicates that it supports login redirection. SAP ASE supports login redirection for RPC requests.

If the initiating server uses site handlers, the cluster workload manager is bypassed and the RPC runs in the system logical cluster of the instance that accepted the connection.

6.7.2 RPCs Where the Local Server is a Cluster

The SAP ASE Cluster Edition does not support outbound RPCs through site handlers, and uses CIS RPC handling by default. Because outbound RPCs from a cluster are identified using the name of the cluster, configure remote servers to accept RPCs from the cluster, not from the individual instances.

6.7.3 RPCs Where Local and Remote Servers Are Instances in the Same Cluster

Instances in a cluster occasionally use RPCs for intra-cluster communication.

When the cluster is started, SAP ASE automatically adds each cluster instance to the `sysservers` table, and removes any cluster instances from `sysservers` that are no longer in the cluster definition. This is also done when instances are dynamically added and dropped at runtime. These `sysservers` entries have the `cluster instance` status bit set. Because intra-cluster RPCs are intended for specific instances, the `sysservers` entries do not set the `enable login redirection` status bit.

6.7.4 sp_serveroption

The `sp_serveroption` system procedure includes the `enable login redirection` and `cluster instance` options.

enable login redirection

`sp_serveroption enable login redirection` determines whether incoming RPC requests may be sent to another instance in the cluster:

```
sp_serveroption <instance_name>, 'enable login redirection', [ true | false ]
```

where:

- `<instance_name>` – is the name of the instance for which you are setting `enable login redirection`. This instance must be included in `sysservers`, and is automatically included for all instances.
- `true` – means the instance can redirect incoming RPC requests to another instance in the cluster.
- `false` – means the instance cannot redirect RPC requests.

By default, `enable login redirection` is enabled, but you must have `sa_role` to run `sp_serveroption`.

cluster instance

`cluster instance` identifies `sysservers` entries that store instance information, where `<instance_name>` is the name of the instance you are adding.

```
sp_serveroption <instance_name>, 'cluster instance', [ true | false ]
```

By default, `<cluster_instance>` is disabled (set to `false`) for each remote server.

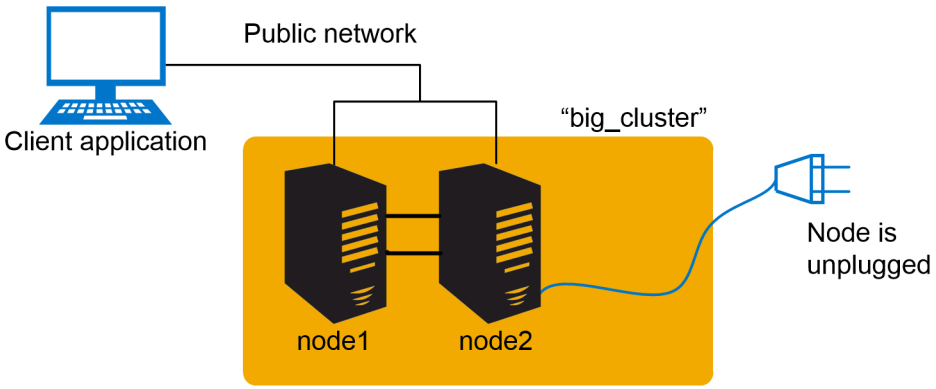
The SAP ASE Cluster Edition automatically manages the `sysservers` rows for instances in the local cluster. You need not manually set or clear the `cluster instance` flag.

6.8 Reconnecting Clients When a Node Loses Power

If the network cable is removed from a machine or a node to which a client is connected loses power, the client-side socket becomes unreachable. The client socket waits, without results, for a reply from the server or for the cluster to issue a send operation.

Context

In the unplugged node scenario illustrated in the following figure, a client application is connected to "big_cluster", which consists of "node1" and "node2" on which instances "ase1" and "ase2" are running, respectively. A client application is connected to instance "ase1" running on "node1".



If the power is disconnected from "node1", the client application waits for contact from the node. The only way to avoid this situation is to configure the client application to assume that the node is down after a specified amount of time, so that it connects to another node in the cluster.

The operating system network detects a crash, disconnects the clients, and fails over the sockets from the remote side of the connection.

To reduce the time required to detect when a cluster loses a host or when a public network is disconnected from a node running an instance:

- Set TCP `keepalive` to a reasonable value on the host on which the client is running.
- Set the client application's `timeout` value.

Procedure

1. Set TCP `keepalive` to a shorter value.

   The TCP `keepalive` parameter eventually marks the client socket as failed. However, because the default value of the TCP `keepalive` value is a long amount of time (some systems may set this to as long as two hours), it may be three or more hours before the client-side sockets fail over. Setting `keepalive` to a small value (several minutes) may not be practical for large organizations; set `keepalive` to an appropriate period of time for your site that works with the HAFAILOVER capabilities.

Set TCP `keepalive` on client machines. The appropriate values depend on your operating system. See your client's operating system documentation for more information.

If you are testing for client timeouts, set the values for the parameters in the first two columns of the following table to a few minutes, and set the values for the parameters in the third column to a low number.

Table 1: Setting TCP keepalive

Operating System	Amount of Time Parameter Waits Before Probing the Connection	Amount of Time Between Probes for Parameter	Maximum Amount of Time or Attempts for Parameter to Probe Connections Before Dropping Them
Linux	`tcp_keepalive_time` Measured in seconds	`tcp_keepalive_intvl` Measured in seconds	`tcp_keepalive_probes` Measured as absolute number

2. Set client's timeout value. There are two different timeout properties you set for Client-Library program connections:
   - CS_LOGIN_TIMEOUT – determines how long the client waits to connect to an unreachable host.
   - CS_TIMEOUT – determines how long a client waits for commands to complete.

   Based on how you configure the timeout event, the client either fails, or fails over to another node.

   Configure clients to set the Client-Library `CS_TIMEOUT` parameter to determine how long to wait before they time out.

   Set the `CS_TIMEOUT` and `CS_LOGIN_TIMEOUT` parameters or the `isql -t` and `-l` parameters for clients to fail over during a sudden loss of power to the node.

   For more information about the Client-Library parameters `CS_TIMEOUT` and `CS_LOGIN_TIMEOUT`, see the *SAP Open Client Client-Library/C Reference Manual*.

   For information about using the `isql -t` and `-l` parameters, see the *SAP ASE Utility Guide*.

Related Information

Open Client and OCS Version Support [page 72]

7 Using Backup Server in a Clustered Environment

You can use one or more Backup Servers in a clustered environment.

7.1 Nodes Joining Cluster During a Database or Transaction Dump

Nodes – and the instances running on them – can join or leave the cluster while another instance runs `dump database` or `dump transaction`.

Versions of the SAP ASE Cluster Edition earlier than 15.0.3 did not allow nodes to join or leave the cluster during a database or transaction log dump. Any instances started during a dump waited until the dump finished before completing its start-up, and instances waited until the dump were complete before shutting down.

Nodes can join and leave the cluster only during phase one of `dump database`. The phases are displayed during the `dump database` command:

```
Backup Server: 3.43.1.1: Dump phase number 1 completed.
Backup Server: 3.43.1.1: Dump phase number 2 completed.
Backup Server: 3.43.1.1: Dump phase number 3 completed.
```

If a node attempts to join or leave during phase two or phase three of `dump database`, it waits to join until `dump database` completes.

Nodes may leave a cluster politely during a database dump from a normal shutdown.

Once you configure multiple Backup Servers for the cluster, you need not perform any additional configuration to enable nodes to join or leave the cluster during a `dump database` or `dump transaction`.

7.2 Multiple Backup Servers

SAP ASE Cluster Edition allows a cluster to use multiple Backup Servers using one of three methods.

- Dedicated method – each instance is assigned a specific Backup Server.
- Round-robin method – at the time of the `dump` or `load` command, the SAP ASE Cluster Edition assigns the instance a Backup Server from a group according to availability.
- A single Backup Server named SYB_BACKUP

The SAP ASE Cluster Edition stores information about all Backup Servers in `sysservers`. When you issue a `dump` or `load` command from any instance, the SAP ASE Cluster Edition searches `sysservers` for the

Backup Server entry named SYB_BACKUP. If it finds the SYB_BACKUP entry pointing to a `srvnetname` entry of $DEDICATED or $ROUNDROBIN, it uses the appropriate Backup Server to perform the `dump` or `load` operation.

For example, the "snack" cluster is configured to use the dedicated method, and includes an instance named "cupcake". Because the SYB_BACKUP entry points to "$DEDICATED", the cluster uses the cupcake_BS Backup Server to perform any `dump` or `load` operation:

```
srvid srvstatus  srvname    srvnetname  srvclass srvsecmech srvcoss  srvstatus2
----- ---------  ---------- ----------- -------- ----------  ------- ----------
    1         8  SYB_BACKUP  $dedicated        7       NULL    NULL           2
    2         8     cupcake     cupcake        7       NULL    1000           4
    3         8  cupcake_BS  cupcake_BS        7       NULL       0           2
    4         8      cookie      cookie        7       NULL    1000           4
    5         8   cookie_BS   cookie_BS        7       NULL       0           2
```

If the snack cluster was configured for a round-robin method, the cupcake instance would use the first available Backup Server to perform `dump` and `load` operations.

When you perfrm an upgrade, the SAP ASE Cluster Edition adds entries for all Backup Servers to `sysservers`.

Entries for Backup Servers are removed from `sysservers` when you remove an instance from the cluster.

7.2.1 Configuring SAP ASE Cluster Edition to Use Multiple Backup Servers

Add additional Backup Servers using `sp_addserver`.

```
sp_addserver <instance_name>_BS, NULL, <instance_name>
```

For example, this adds the cookie_BS entry to `sysservers`:

```
sp_addserver cookie_BS, NULL, cookie
```

Insert interfaces file entries for each of the Backup Servers before you can start them.

7.2.1.1 Using the Dedicated Method

Configure the SAP ASE Cluster Edition for a dedicated method using `sp_addserver 'SYB_BACKUP', NULL, '$DEDICATED'`

Each instance in a cluster configured for a dedicated Backup Server method must include a Backup Server.

When a user connects to instance "cookie" using `isql` and issues a `dump` or `load` command, instance cat finds the SYB_BACKUP entry in `sysservers` pointing to the "$DEDICATED" keyword. If the specified Backup Server ("cookie_BS") is not running, the SAP ASE Cluster Edition automatically restarts this Backup Server.

7.2.1.2 Using the Round-Robin Method

Configure the SAP ASE Cluster Edition for a round-robin method using `sp_addserver 'SYB_BACKUP', NULL, '$ROUNDROBIN'`.

Once you configure the SAP ASE Cluster Edition to use the round-robin method, the SAP ASE Cluster Edition selects a Backup Server for a `dump` or `load` command according to which Backup Server is performing the least amount of work.

For example, if a user connects to instance Inst1 from `isql` and issues a `dump` command from this session, instance Inst1 checks the SYB_BACKUP entry in `sysservers`, which points to keyword $ROUNDROBIN. If Inst1_BS is busy performing another task, the SAP ASE Cluster Edition moves to the next Backup Server, Inst2_BS. If Inst2_BS is available, with no jobs running on it, this Backup Server performs the dump, and its index is incremented to indicate that it is busy and unavailable for work. However, if Inst2_BS is busy, the SAP ASE Cluster Edition moves to the next Backup Server, and so on, until it finds an available Backup Server.

If, on the other hand, SAP ASE Cluster Edition cannot connect to Inst2_BS, then the `enable backupserver HA` parameter determines whether to attempt to connect to the next Backup Server, or to restart Backup Server automatically. An error is reported only if all configured Backup Servers are unavailable.

If you set `enable backupserver HA` to `true` and configure Backup Server for a round-robin policy, you must enable all nodes to start any Backup Server in the cluster. This allows another node to perform the `dump` or `load` duties if the initial node fails on which you issued the command. To enable all nodes to start any Backup Server, include entries for all Backup Servers in the `interfaces` file.

For example, Big_Cluster uses a round-robin policy, and includes instances Node_1, Node_2, and Node_3. The `interfaces` file entry for instance Node_1 must include entries for Backup Servers running on nodes Node_2 and Node_3 in case the round-robin policy select either of these as the replacement node if Node_1 fails:

```
Node_1_BS
     master tcp ether Node_1 5004
     query tcp ether Node_1 5004
     master tcp ether Node_2 5006
     query tcp ether Node_2 5006
     master tcp ether Node_3 5008
     query tcp ether Node_3 5008
Node_2_BS
     master tcp ether Node_1 5004
     query tcp ether Node_1 5004
     master tcp ether Node_2 5006
     query tcp ether Node_2 5006
     master tcp ether Node_3 5008
     query tcp ether Node_3 5008
Node_3_BS
     master tcp ether Node_1 5004
     query tcp ether Node_1 5004
     master tcp ether Node_2 5006
     query tcp ether Node_2 5006
     master tcp ether Node_3 5008
     query tcp ether Node_3 5008
```

Alternatively, you can manually start Backup Server on the node from which you perform the `dump` or `load` operation.

7.2.2 Starting and Stopping Backup Servers

`sybcluster` and the SAP Control Center automatically start Backup Servers after creating them.

For information about starting and stopping Backup Servers with `sybcluster`, see the *SAP ASE Utility Guide*. For information about Starting Backup Server with the SAP Control Center, see *SAP Control Center for ASE*.

- Start Backup Servers manually using the `startserver` utility. `startserver` requires that Backup Server have a `runserver` file. This example starts the MOUSE_BS Backup Server:

  ```
  startserver -f RUN_MOUSE_BS
  ```

  For more information about the `startserver` utility, see the *SAP ASE Utility Guide*.
- Shut down a specific Backup Server using the `shutdown <server_name>`.
  For example:

  ```
  shutdown MOUSE_BS
  ```
- Shut down all active Backup Servers using `shutdown SYB_BACKUP`.

7.2.3 Backing Up to Media

The process of backing up objects for the SAP ASE Cluster Edition is the same as in a nonclustered SAP ASE.

See the *SAP ASE System Administration Guide, Volume 2*.

7.2.4 Change Backup Servers Using Stored Procedures

You can make changes to stored procedures using `sp_addserver`, `sp_helpserver`, `sp_dumpoptimize`, and `sp_volchanged`.

- `sp_addserver` adds the keywords $DEDICATED and $ROUNDROBIN as a parameter for setting the Backup Server method.
- When the SYB_BACKUP entry in `sysservers` points to the keywords $DEDICATED or $ROUNDROBIN, the SAP ASE Cluster Edition runs`sp_dumpoptimize` and `sp_helpserver` for all active Backup Servers in the cluster.
- `sp_volchanged` allows you to include the name of a specific Backup Server.
  The syntax is:

  ```
  sp_volchanged <session_id>, <devname>, <action> [, <fname> [, <vname>] [,
  <backup_server_name>]]
  ```

  For example, this specifies that a volume change request is made on Backup Server Inst3_BS:

  ```
  sp_volchanged 5, "db1.dmp", "proceed", "Inst3_BS"
  ```

  If you do not include the name of a Backup Server with the $DEDICATED or $ROUNDROBIN method, `sp_volchanged` displays an error message.

8 Run Job Scheduler in a Clustered Environment

The SAP ASE Cluster Edition allows you to run the Job Scheduler in a shared-disk cluster environment. A single Job Scheduler services all job scheduling requests that originate from all instances in the cluster. Job Scheduler runs on the boot coordinator.

If you manually start the Job Scheduler after the cluster is started, the cluster selects the coordinator to host the Job Scheduler.

For information about installing, starting, and configuring Job Scheduler, see the *SAP ASE Job Scheduler Users Guide*.

8.1 Installing and Configuring Job Scheduler

The coordinator instance running the Job Scheduler is also known as the "Job Scheduler instance".

To set up and start the Job Scheduler, see the *Job Scheduler Users Guide*. For instructions for installing and configuring Job Scheduler in the SAP ASE Cluster Edition, see the installation guide for your platform.

8.2 Running Job Scheduler in a Clustered Environment

If you perform a Job Scheduler action on an instance other than the Job Scheduler instance, the instance issues a request to the Job Scheduler instance. The Job Scheduler processes the request and sends the reply back to the requesting instance.

8.3 Shutting Down Job Scheduler

Shut down the Job Scheduler in a shared-disk cluster using the same steps as in a nonclustered SAP ASE configuration.

Procedure

1. From any instance, right-click the Scheduled Jobs folder, which appears under each instance; each instance can initiate all Job Scheduler commands from the SAP Control Center.

2. Select **Administer** to open the **Job Scheduler Administration** dialog.
3. From the **Task Configuration** tab, select **Stop**, or from the command line, run:

```
use sybmgmtdb
go
sp_js_wakeup "stop_js", 1
go
```

8.4 Redirecting Scheduled Jobs

When the Job Scheduler agent attempts to connect to an instance to execute a scheduled job and the instance is too busy, the Workload Manager may redirect the connection to another instance in order to allow any instance in the cluster to perform the scheduled jobs. This happens automatically; you need not reconfigure Job Scheduler.

The Workload Manager takes care of login redirection. You can affect how connections are redirected by setting rules at the logical cluster level. However, if you do not want a scheduled job to be redirected, set a new job property that allows redirection:

- From SAP Control Center – unselect the **Allow Redirection** option in the **Job Properties** dialog box. The default behavior allows the Workload Manager to redirect a scheduled job connection according to the workload rules that exist at the time the connection is made.
- From the command line – set the redirection property `no_conn_redirection` when you create or modify a job. This example sets the property for a job named `find_old_logins` using `sp_sjobcreate`:

```
sp_sjobcreate @name='jname=find_old_logins', @option='jcmd=exec
sp_find_old_logins,jproperties=no_conn_redirection=true'
```

9 Using Java in a Clustered Environment

SAP ASE Cluster Edition supports third-party Java Virtual Machines (JVMs), such as the Sun Java 2 Platform, Standard Edition (J2SE), in the clustered environment.

To install the Java components on the SAP ASE Cluster Edition, see the installation guide for your platform. For a complete description of how to use Java on SAP ASE, see *Java in SAP ASE*.

In general, you can use Java in the same way in clustered and nonclustered environments. However:

- In the SAP ASE Cluster Edition, connection migration is not supported for connections executing Java. If a connection fails, a connection executing Java fails over to another instance as configured. After failover, existing SQL batch processes and Java are executed again automatically.
- SAP ASE handles static data in different ways, depending on whether or not the JVM is shared or private.
  - In the SAP ASE Cluster Edition, each node has its own JVM – static data is not shareable between nodes. If a node goes down, the JVM goes down as well, and all work on the JVM is lost. You must reapply any extant SQL batch processes.
  - The nonclustered SAP ASE shares a single JVM across all engines and connections – static data is sharable for classes loaded from a common class loader.
  - In SAP ASE 15.0.2 and earlier, each client connection has its own private Sybase JVM – static data is not supported across clients.

10 Using System Tables in the Cluster Edition

There are some differences in system tables if you are using a Cluster Edition of SAP ASE.

10.1 timestamp Columns

In the SAP ASE server, if a table includes a `timestamp` column, its value is updated when a row is changed. Client applications can use this functionality to detect changes to rows using an access method called "optimistic locking."

The values in the `timestamp` column are unique in a database. However, in the Cluster Edition, `timestamp` column values are not guaranteed to be in increasing order in a database across tables, but they are guaranteed to be in increasing order for a particular table.

10.2 Changed Identity Values

Identity columns in the Cluster Edition behave differently from those in non-clustered editions of SAP ASE. Although the Cluster Edition guarantees that identity values are unique, for performance reasons the values may not monotonically increase.

In a non-clustered SAP ASE server, a set of identity values are burned into memory to reduce disk I/Os as `inserts` access the next value from memory. In the Cluster Edition, the same size `set` is burned into memory, but the `set` is split among the cluster instances. In a two-instance cluster with an identity `set` size of 250000, the first instance inserts values {1,2,3, and so on}, and the second instance inserts values {125000,125001,125002, and so on}.

When you restart an instance after a shutdown or a crash, a new block of identity values may be allocated for that instance when it rejoins the cluster. As a result, you may see a jump in the identity value for the instance.

The `next-identity` function reports the next identity value for a table from the instance in which `next-identity` is executed. For example, `next-identity` returns 4 for instance 1 and 125003 for instance 2.

The behavior of the `identity-burn-max` remains the same as for a non-clustered SAP ASE server because the burn size and burn behavior is unchanged in the Cluster Edition.

10.3 Controlling Fake-Table Materialization

Certain stored procedures, such as `sp_who` and `sp_lock`, read from fake tables such as `sysprocesses` and `syslocks`. Because their rows are not stored on disk, fake tables present an exception to the shared-data nature of a shared-disk cluster, and special features apply.

You can control whether a fake-table query returns rows from the local instance or all instances in the cluster by using the `set system_view` command. `set system_view` is a session-level command. In addition, `set system_view` also controls monitoring table materialization.

By default, the SAP ASE server retrieves rows only from the local instance.

- To specify that fake-table queries materialize rows for all instances, use the `cluster` option. For example:

  ```
  set system_view cluster
  ```

- To specify that fake-table queries materialize rows for the local instance, use the `instance` option. For example:

  ```
  set system_view instance
  ```

To retrieve the current `system_view` setting, select the `<@@system_view>` global variable.

The SAP ASE server supports cluster-wide materialization for these fake tables:

- `sysprocesses`
- `syslocks`
- `sysengines`
- `syslisteners`
- `sysmonitors`
- `syssecmechs`
- `syscurconfigs`

> **Note**
>
> `sysinstances` is always set for cluster-wide materialization, regardless of the `system_view` setting.

11 Monitoring Tables for Clusters

In a clustered environment, monitoring tables report on a per-instance basis instead of returning a cluster-wide result, allowing you to monitor the activities of processes and queries across the cluster to get a better understanding of the statistics for objects that may be opened on more than one instance and resource usage on each instance in the cluster.

For example, if you query the monitoring tables about a table, this table may be opened or accessed by more than one instance in the cluster, so the descriptors for this table – and the associated statistics – may be in memory on the instance. Statistics are not aggregated for the cluster. The statistical results for all instances are returned as a unioned result set with rows collected from each instance. Each instance is identified in the result set with a row in the `InstanceID` column.

For column and datatype descriptions for these monitoring tables, see the *SAP ASE Reference Manual: Tables*.

11.1 Configuring the System View

Configure system view using `set system_view` command.

Context

The `system_view` is a session-specific setting that controls the scope of monitoring data that queries return from the monitoring tables, `sysprocesses`, `sp_who`, and other commands. When you set `system_view` to `cluster`, queries on the monitoring tables return data from all active instances in the cluster. When you set `system_view` to `instance`, queries on the monitoring tables return data only for processes or objects that are active on the instance to which the client is connected.

Procedure

Use the `set` command to configure the scope of the session:

```
set system_view {instance | cluster | clear}
```

where:

- `instance` – returns statistics for the local instance only. Cross-cluster requests are not sent to any other instance in the cluster.
- `cluster` – returns statistics for all instances in the cluster.
- `clear` – returns the system view to the configured default.

This example modifies the session settings so that queries on the monitoring tables return data only for the instance to which the client is connected:

```
set system_view instance
```

This example modifies the session settings so that queries on the monitoring tables return data for the cluster:

```
set system_view cluster
```

This example clears the current setting of the system view and returns it to the default setting.

```
set system_view clear
```

If you do not specify an `InstanceID` when you query a monitoring table or call a monitoring table RPC, the instance uses the current `system_view` configuration.

The session system view is inherited from its host logical cluster. Select the `<@@system_view>` global variable to determine the current system view.

11.2 Configuring Monitoring Tables

Determine cluster-wide and instance-only behavior using configuration parameters for the monitoring tables. By default, all monitoring table configuration values are applied cluster-wide.

11.3 Message Pipe Parameters

Message pipe parameters determine how the cluster and instances manage the memory used to store the data for the historical monitoring tables.

The parameters are:

- `deadlock pipe max messages`
- `errorlog pipe max messages`
- `sql text pipe max messages`
- `plan text pipe max messages`
- `statement pipe max messages`

Configure these parameters globally for the cluster and individually for each instance. These parameters allocate memory for the pipe. An instance can dynamically add memory to the pipe but cannot dynamically remove memory from the pipe, so if you reduce the size of the parameter, restart the instance for the new pipe size to take effect.

These algorithms determine the size for the parameter:

- For an individual instance, the memory required for the each pipe configuration is: `<configuration_value>` X `<number_of_engines>`

- To globally set the memory for each pipe configuration:
  <configuration_value> X <number_of_engines> X <number_of_instances>
- If you set the value for pipe configurations differently for each instance, the amount of memory required for the cluster is:
  (<instance_1_value> X <number_of_engines>) + (<instance_2_value> X <number_of_engines>) + . . . + (<instance_n_value> X <number_of_engines>))

11.4 Changes for RPCs

If you invoke an RPC but do not include the `InstanceID` as a parameter, the monitoring tables use the system view setting to determine how to report the statistics.

If you set the system view setting to `instance`, all data gathering is local. If you set the system view setting to `cluster`, the monitoring tables communicate with instances that form the instances in the cluster, not the logical cluster.

11.5 Monitoring Tables Specifically for the Cluster Edition

These monitoring tables are specifically for the SAP ASE Cluster Edition.

- `monClusterCacheManager`
- `monCIPC`
- `monCIPCEndpoints`
- `monCIPCLinks`
- `monCIPCMesh`
- `monCLMObjectActivity`
- `monCMSFailover`
- `monDBRecovery`
- `monDBRecoveryLRTypes`
- `monFailoverRecovery`
- `monLogicalCluster`
- `monLogicalClusterAction`
- `monLogicalClusterinstance`
- `monLogicalClusterRoute`
- `monPCM`
- `monProcessMigration`
- `monTempdbActivity`
- `monWorkloadPreview`
- `monWorkloadProfile`
- `monWorkloadRaw`

11.6 Monitoring Tables that Return Identical Information for All Instances

These monitoring tables include the same information for all instances.

Table 2: Monitoring Tables That Include the Same Information for All Instances

Table name	Description
monMon	Metadata view is identical on all instances.
monTableColumns	Metadata view is identical on all instances.
monTableParameters	Metadata view is identical on all instances.
monTables	Metadata view is identical on all instances.
monWaitClassInfo	List of descriptions is identical on all instances.
monWaitEventInfo	List of descriptions is identical on all instances.

11.7 Monitoring Tables that Return Specific Instance Information

These monitoring tables contain the InstanceID column.

- monCachePool
- monCachedObject
- monCachedProcedures
- monDataCache
- monDeadLock
- monDeviceIO
- monEngine
- monErrorLog
- monIOQueue
- monLicense
- monLocks
- monNetworkIO
- monOpenDatabases
- monOpenObjectActivity
- monOpenPartitionActivity
- monProcedureCache
- monProcess

- monProcessActivity
- monProcessLookup
- monProcessNetIO
- monProcessObject
- monProcessProcedures
- monProcessSQLText
- monProcessStatement
- monProcessWaits
- monProcessWorkerThread
- monResourceUsage
- monState
- monSysPlanText
- monSysSQLText
- monSysStatement
- monSysWaits
- monSysWorkerThread

12 Managing the Workload

The SAP ASE Cluster Edition workload manager can customize workload management and failover for each of your business applications so that each performs most efficiently. The logical cluster is the container that allows the workload manager to provide individualized working environments.

A logical cluster is an abstract representation of one or more instances in a physical shared-disk cluster. Each logical cluster has a set of instances on which it runs on and can have a set of instances to which it fails over. Routing rules direct incoming connections to specific logical clusters based on an application, user login, or server alias supplied by the client. Other rules can restrict a logical cluster to bound connections or allow any authenticated connection to access it.

By creating logical clusters on top of the physical cluster, you can partition applications with different workloads on the same system. The workload is managed at the application level, so that you can manage incoming connections, failover policies, load distribution strategies, and planned downtime according to how each of your applications use the system.

The system administrator manages the workload using SAP Control Center or the command line options. The system administrator:

- Configures and manages a logical cluster. This includes creating and dropping logical clusters, adding or removing instances from the cluster, modifying failover rules, starting and stopping the cluster or instances in the cluster, configuring routing rules, and so on.
- Selects or configures load profiles that the system uses to determine the current relative workload.
- Monitors instances in the cluster and the workload on each instance.

12.1 Logical Cluster Resources

A logical cluster is assigned resources from the physical cluster.

- Instances – are a logical cluster's base instances. They are started when a logical cluster starts and that a failback restores them.
- Failover resources – are an ordered list of instances on which a logical cluster is to run if one or more of the base instances fail. Any instance in the physical cluster can be a failover resource. Workload management capabilities let you group and configure resources to specify failover order and precedence.

12.2 The System Logical Cluster

When you create a shared-disk cluster, SAP ASE Cluster Edition automatically creates a system logical cluster for it.

The system logical cluster provides a logical cluster representation for daemons and enables management of certain tasks. It represents the physical cluster, contains all the instances of the physical cluster, and has the

same name as the physical cluster. All background tasks, such as checkpoint and housekeeper, run on the system logical cluster. Special rules apply to the system logical cluster. Upon creation, this new system logical cluster is granted the open property, so that all unbound connections are routed to it.

System administrators typically do not interact with the system logical cluster. However, the following do apply to the system logical cluster:

- Routing rules. For example, you can route logins used by system administrators to the system logical cluster.
- The open property.
- System view setting.
- The load profile.
- Login distribution mode.

The following do not apply to the system logical cluster:

- Failover resources and rules
- Commands that:
  - Create or drop resources of a logical cluster.
  - Migrate, fail over, and fail back instances.
  - Change a logical cluster or instance state.
  - Change failover mode settings.
  - Change start-up mode settings.
  - Set the down routing mode.

12.3 Configuring a Logical Cluster

There are many possible options for configuring a logical cluster.

The basic steps for setting up a working logical cluster are:

1. Create a logical cluster.
2. Add instances.
3. Assign routing rules.
4. Start a logical cluster.

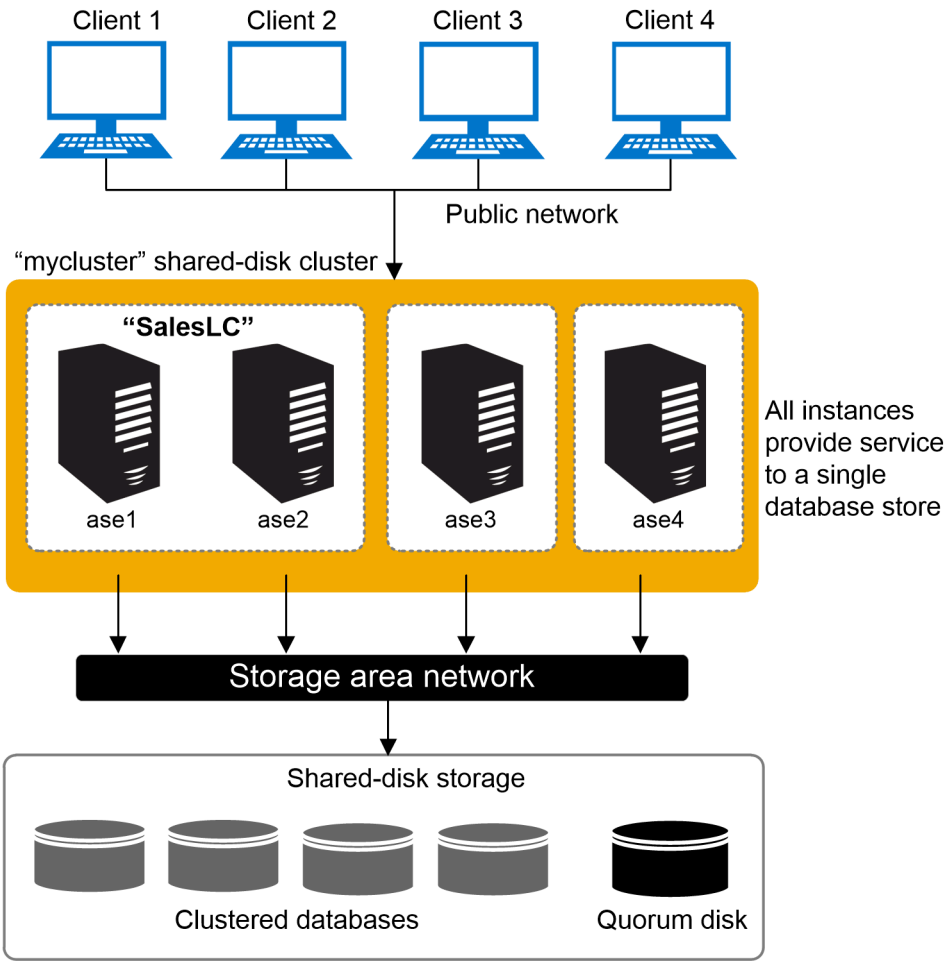
12.3.1 Creating a Logical Cluster

Create a logical cluster using `sp_cluster logical, "create"`.

In the example illustrated in the figure, we create three logical clusters for the "mycluster" physical cluster: "SalesLC", "HRLC", and "CatchallLC". A physical cluster called "mycluster" contains four instances: "ase1", "ase2", "ase3", and "ase4". We create three logical clusters:

- "SalesLC" – to handle applications and logins from the Sales Department.
- "HRLC" – to handle applications and logins from the Human Resources Department.

- "CatchallLC" – to use later for an open logical cluster.



To create "SalesLC", "HRLC", and "CatchallLC", enter:

```
sp_cluster logical, "create", SalesLC
sp_cluster logical, "create", HRLC
sp_cluster logical, "create", CatchallLC
```

You can also use the SAP Control Center to create logical clusters. For more details, see SAP Control Center documentation

12.3.2 Adding Instances to a Logical Cluster

Add instances to a cluster using `sp_cluster logical, "add"`.

- This example adds two instances to the mission-critical "SalesLC":

```
sp_cluster logical, "add", SalesLC, instance, ase1
sp_cluster logical, "add", SalesLC, instance, ase2
```

- This example adds a single instance to "HRLC":

```
sp_cluster logical, "add", HRLC, instance, ase3
```

- This example adds a single instance to "CatchallLC":

```
sp_cluster logical, "add", CatchallLC, instance, ase4
```

12.3.3 Adding Routes to a Logical Cluster

Route clients to a target logical cluster using `sp_cluster logical, "add"`.

- This example routes the applications "field_sales" and "sales_reports" to "SalesLC":

```
sp_cluster logical, "add", SalesLC, route, application,
"field_sales;sales_reports"
```

- This example routes the login name "sales_web_user", which is used by several sales applications via the Internet, to "SalesLC":

```
sp_cluster logical, "add", SalesLC, route, login,
sales_web_user
```

- This example routes all clients using human resources applications to an alias "HRLC":

```
sp_cluster logical, "add", HRLC, route, alias,
HR_SERVER
```

> **Note**
>
> Make sure to include each server alias in the client's directory service, and that those query entries specify an address on which the physical cluster is listening.

For example, to set up an alias for the client's directory service, use:

```
ase1
     query tcp ether blade1 19786
ase2
     query tcp ether blade2 19786
ase3
     query tcp ether blade3 19786
ase4
     query tcp ether blade4 19786
mycluster
     query tcp ether blade1 19786
     query tcp ether blade2 19786
     query tcp ether blade3 19786
     query tcp ether blade4 19786
HR_SERVER
     query tcp ether blade1 19786
     query tcp ether blade2 19786
     query tcp ether blade3 19786
     query tcp ether blade4 19786
```

For complete `sp_cluster` syntax and usage information, see the *SAP ASE Reference Manual: Procedures*.

Related Information

Routing Rules [page 102]

12.3.4 Starting a Logical Cluster

Start a logical cluster using `sp_cluster logical, "online"`. This places it in the online state.

This example starts "SalesL" and "HRLC":

```
sp_cluster logical, "online", SalesLC
sp_cluster logical, "online", HRLC
```

12.4 Routing Rules

Routing rules direct incoming client connections to the appropriate logical cluster. Once a connection is routed to a logical cluster, a server task entered for that route can be administered by logical cluster management capabilities.

There are three types of routing rules, or bindings. Each binding uses the name in the TDS login record.

Binding routes are listed in the order of precedence, from highest to lowest (so that a login route takes precedence over an application route, which takes precedence over an alias route):

- Login routes – establish a relationship between a server login and a logical cluster.
- Application routes – establish a relationship between an application name and a logical cluster.
- Alias routes – associate a server name with a logical cluster. Applications can choose a logical cluster from server aliases placed in the interfaces file. These aliases use unique server names. The alias entry can point anywhere in the physical cluster. The workload manager sends it to the correct instances via login redirection.

Each client connection to a shared-disk cluster is associated with a logical cluster. That association can be based on a routing rule; it can also be based on the lack of a routing rule, which means the connection is routed to the open logical cluster. If the connection cannot be directed to a logical cluster, either because the target logical cluster is offline or the client does not support login redirection, the connection can be handled according to the target logical cluster's down-routing mode.

12.5 Logical Cluster Attributes

Every logical cluster possesses a set of attributes, or properties, that control different aspects of logical cluster behavior. Each attribute has a default value. You can accept the default value or change it to suit your application environment.

Use `sp_cluster logical, "set"` to manage the following attributes:

- Open – specifies a logical cluster to which clients without a specific routing plan are directed.
- System view – specifies whether monitoring and informational tools such as `sp_who`, `sp_lock`, and monitoring tables describe an instance in the cluster or the whole cluster
- Start-up mode – specifies whether a logical cluster must be started automatically or manually.
- Down-routing mode – specifies how client connections are routed if a logical cluster designated by the routing rule is not available.
- Failover mode – how and when failover instances are brought online.
- Fail-to-any – specifies whether any instance or only a designated instance can be a failover resource.
- Load profile – provides a series of weighted metrics to determine the relative workload on an instance in a logical cluster.
- Login distribution mode – specifies how SAP ASE Cluster Edition distributes connections when a logical cluster spans multiple instances.
- Action release – releases and clears `online`, `offline`, `failover`, and `failback` – either manually or automatically after they are completed or cancelled.
- Gather mode – collects groups of connections to a different logical cluster either manually or automatically when one of these predefined actions occurs – `online`, `add route`, `alter route`, or `drop route`.
- Roles – indicates whether the logical cluster is "system", "open", or "none" if the logical cluster is neither.

Related Information

Viewing Information about a Logical Cluster [page 112]

12.5.1 Open Logical Cluster

All connections not routed to a logical cluster via an explicit routing rule are routed to the current open logical cluster. When you create a new cluster, the system logical cluster is automatically designated as the open logical cluster.

Although you can reset the `open` attribute for another logical cluster, only one open logical cluster can exist per physical cluster.

Use `sp_cluster logical, "set"` to specify a new open logical cluster:

```
sp_cluster logical, "set", <lc_name>, "open"
```

This example designates "CatchallLC" as the open logical cluster:

```
sp_cluster logical, "set", CatchallLC, "open"
```

When you reset the `open` attribute for a new logical cluster, SAP ASE Cluster Edition automatically turns off the open attribute for the old open logical cluster.

Use the open property with the down-routing mode to reserve one or more instances for the exclusive use of a specific logical cluster.

12.5.2 System View Attribute

The `system_view` attribute controls how you see the cluster when using system stored procedures such as `sp_who` and `sp_lock`, or viewing monitoring tables, fake tables, and so on. You can set the `system_view` attribute so that SAP ASE Cluster Edition displays information about the current instance or the physical cluster.

To set the default system view for tasks running in the logical cluster, enter:

```
sp_cluster logical, "set", <lc_name>, "system_view", {instance| cluster}
```

This example sets the system view for the instance in "SalesLC":

```
sp_cluster logical, "set", SalesLC, system_view, instance
```

This example sets system view for the cluster:

```
sp_cluster logical, "set", SalesLC, system_view, cluster
```

Setting the system view at a logical cluster level provides a default value for connections in a logical cluster. See the *SAP ASE Reference Manual: Procedures*.

> **Note**
>
> The `system_view` attribute is an administrative tool. Its current value does not affect the way applications perceive a logical cluster, its databases, or database objects.

12.5.3 Start-up Mode

The start-up mode attribute specifies whether a logical cluster starts automatically when the cluster starts, or whether the administrator starts it manually.

- (Default) In automatic mode, a logical cluster starts when any of its base instances come online after a cluster restart. Failover instances automatically come online if other failover instances are currently online.
- In manual mode, a logical cluster comes online only after the administrator executes the `online` command.

To set the start-up mode of the logical cluster, enter:

```
sp_cluster logical, "set", <lc_name>, "startup", { automatic | manual}
```

This example changes the start-up mode for "SalesLC" to manual:

```
sp_cluster logical, "set", SalesLC, startup, manual
```

12.5.4 Down-Routing Mode

Routing rules direct incoming client connections to the appropriate logical cluster. However, routing rules do not specify how connections should be handled when the target logical cluster is offline, or when redirection is required and the connection does not support redirection.

> **Note**
>
> The Client-Library property CS_PROP_REDIRECT determines whether a client connection supports login redirection. By default, the value of CS_PROP_REDIRECT is true, and the client connection supports login redirection. See the *Open Client Client-Library/C Reference Manual*.

You can specify a down-routing mode to direct connections when routing rules cannot be applied.

You can also use this attribute to reserve certain instances for specific connections.

To set the default system view for tasks running in the logical cluster, enter:

```
sp_cluster logical, "set", <lc_name>, "down_routing", {system | open |
disconnect}
```

Where:

- `system` – (default) sends unroutable connections to the system logical cluster. `system` ensures that the highest availability as the system logical cluster is always online for every instance.
- `open` – sends unroutable connections to the open logical cluster. If the connection cannot be sent to the open logical cluster, or the connection does not support redirection, SAP ASE applies the down-routing mode of the open logical cluster.
- `disconnect` – disconnects unroutable connections. This setting can enforce resource reservation by disconnecting clients that cannot be served by instances in the target logical cluster.

This example sets the down-routing mode to `open` for "SalesLC":

```
sp_cluster logical, "set", SalesLC, down_routing,
"open"
```

This example ensures that, if "SalesLC" is unavailable, clients bound to that logical cluster are routed to the open logical cluster.

Related Information

Routing Rules [page 102]

12.5.4.1 Reserving Resources

Reserve one or more instances for the exclusive use of a specific logical cluster by using the open property with the `disconnect` down-routing mode.

For example, to reserve instance "ase3" for the exclusive use of the "HRLC" logical cluster:

1. Set the open property to a logical cluster that does not include "ase3".
2. Set the down-routing mode of the open logical cluster to `disconnect` so that even clients that do not support redirection cannot access it.

Only connections with routing rules specifying "HRLC" can connect to "ase3".

12.5.5 Failover Mode

The failover mode specifies when and how a logical cluster runs on failover instances.

To set the failover mode of the logical cluster, enter:

```
sp_cluster logical, "set", <lc_name>, "failover", {instance | group}
```

Where:

- `instance` (default) – specifies that whenever an online instance fails (whether a base or failover instance), it is replaced with a failover instance on a 1:1 basis. For example, logical cluster "SalesLC" runs on instances "ase1" and "ase2" with "ase3" and "ase4" as failover instances, and the failover mode is "instance". If "ase1" fails, "ase3" comes online, and the cluster runs on "ase2" and "ase3" (or "ase4", depending on the relative workload of the two failover instances).
- `group` – specifies that base instances are replaced only when all base instances fail and that all failover instances then come online. Suppose the failover mode for "SalesLC" is "group". If "ase1" fails, the cluster continues to run on "ase2". No failover instances are brought online. However, if both "ase1" and "ase2" fail, then the cluster runs on "ase3" and "ase4".

You can also designate multiple failover groups, so that even if instances in the first failover set fail, another set of failover instances is available to come online.

This example sets the failover mode for "SalesLC" to "group":

```
sp_cluster logical, "set", SalesLC, failover, "group"
```

12.5.6 fail_to_any Attribute

The `fail_to_any` attribute determines whether a logical cluster can fail over to any instance in the cluster, or only to designated failover instances. This attribute becomes important only if designated failover instances cannot be brought online.

To set a logical cluster to fail over to any instance in the cluster, or only to designated failover instances, enter:

```
sp_cluster logical, "set", <lc_name>, "fail_to_any", {true | false}
```

Where:

- `true` – specifies that the system always selects other instances to act as failovers as long as any other instances in the cluster are online and available. This is the default value.
- `false` – specifies that only designated failover instances can be used.

This example turns off the `fail_to_any` attribute for "SalesLC":

```
sp_cluster logical, "set", SalesLC, fail_to_any, false
```

12.5.7 Load Profile Attribute

SAP ASE uses the load profile to provide a load score for each instance in a logical cluster. The load score determines when the workload manager directs connections to other instances to help balance the workload.

Use the sample load profiles tested and provided by SAP or configure your own.

To set a load profile, use:

```
sp_cluster 'logical', 'set', <lcname>, 'load_profile', <profile_name>
```

This example sets the load profile for the "SalesLC" logical cluster to the SAP ASE profile ase_profile_oltp:

```
 sp_cluster logical, "set", SalesLC, load_profile, ase_profile_oltp
```

Related Information

Load Profiles [page 124]

12.5.8 Login Distribution Mode

The login distribution mode lets you specify how connections are distributed in logical clusters with multiple instances. The login distribution mode does not affect single-instance logical clusters.

To specify how the Cluster Edition distributes connections when a logical cluster spans multiple instances, enter:

```
 sp_cluster logical, "set", <lc_name>, login_distribution, { affinity | round-
robin }
```

Where:

- `affinity` – specifies that the instance accepting a connection retains it as long as the target logical cluster is running on that instance.
  - If the load profile specifies a load threshold, and the load on the instance is too high, the workload manager redirects the connection to the least loaded instance in a logical cluster.

- If the target logical cluster is not running on the instance, the workload manager redirects the connection to the least loaded instance in a logical cluster.

- `round-robin` – specifies that incoming connections are distributed in a round-robin fashion among the instances that are hosting a logical cluster. For example, if "SalesLC" is running on "ase1" and "ase2", the workload manager sends the first connection to "ase1", the next to "ase2", and so on. Load scores are not included in the algorithm.

> **Note**
>
> The SAP ASE Cluster Edition does not perform load-based redirection (affinity or round-robin) for connections with the sa_role. However, sa_role connections are redirected if a route directs them to a logical cluster running on another instance.

Use affinity mode for transactional applications with short, frequent connections, and round-robin mode for read-mostly applications where an application server establishes a persistent pool of connections to the SAP ASE Cluster Edition.

12.5.9 Action Release

Action release clears all logical cluster actions – `online`, `offline`, `failover`, and `failback` – either manually or automatically after these actions are completed or cancelled.

To enable or disable automatic clearing of logical cluster actions, enter:

```
sp_cluster logical, "set", "<lc_name>", "action_release",
    {"automatic" | "manual"}
```

To manually clear all actions for a logical cluster, enter:

```
sp_cluster logical, "action", "<lc_name>", "release", "all"
```

12.5.10 Gather Mode

Gather mode lets you gather all qualified connections on the system, or those connecting to an open logical cluster, and move them to another logical cluster using defined routing rules.

Gather mode can automatically move the connections when a predefined action, such as `online`, `add route`, `alter route`, or `drop route`, occurs, or you can gather connections manually. SAP ASE Cluster Edition then looks for all connections that match the routing rules for the affected logical cluster, and migrates them to the specified logical cluster.

To manually gather the connections and move them to a named logical cluster, enter:

```
sp_cluster logical, "gather", "<lc_name>"
```

To enable or disable the automatic gathering of connections, enter:

```
sp_cluster logical, "set", "<lc_name>", "gather", {"automatic" | "manual" }
```

Related Information

Migrating Connections [page 115]

12.5.11 Roles

The logical cluster role values are "system", "open", and "none".

- System – indicates that that named logical cluster is the system logical cluster.
- Open – indicates that the named logical cluster is the logical cluster to which clients are directed if they do not have a specific routing plan.
- None – indicates that the logical cluster is neither the system nor the open logical cluster.

To view the current role of the logical cluster, enter:

```
sp_logical cluster, "show", "<lc_name>"
```

Related Information

The System Logical Cluster [page 98]
Open Logical Cluster [page 103]

12.6 Configuring Failover

Any instance in a physical cluster can be a failover resource. Logical cluster failover rules do not impact infrastructure, lock remastering, or recovery.

Configure logical cluster failover by specifying:

- Failover resources – specific instances or groups of instances to which failover occurs.
- The failover mode – determines whether failover occurs for individual members of a logical cluster or only for the cluster as a whole.
- The `fail_to_any` attribute – determines whether failover occurs only to designated failover resources or to any other instance if the designated failover resources are unavailable.

When you create a logical cluster, default settings for the `fail_to_any` attribute ensure that a base instance that fails is immediately replaced with another instance. This is the simplest failover strategy, and is adequate for many sites.

If your site requires finer control of failover resources, change the default settings and direct failover to specific instances or groups of instances.

You can create up to 31 failover groups for each logical cluster. By grouping failover instances, you can give preference to specific failover groups. For example, you can ensure that instances in group 1 are considered

before instances in group 2, and so on. The workload manager chooses failover instances within the group according to workload: instances with the lighter load are chosen first.

Instances can be a member of only one failover group per logical cluster. Thus, if instance "ase4" is in failover group 1 for "SalesLC", it cannot also be in failover group 2 for "SalesLC". However, "ase4" can simultaneously be in failover group 1 for "SalesLC", failover group 2 for "HRLC", and a base instance for "CatchallLC".

When the workload manager needs to activate failover instances, it looks first in group 1, then in group 2, and so on until the failover condition is satisfied. If it cannot activate a configured failover resource, the workload manager checks the `fail_to_any` parameter configuration.

If `fail_to_any` is:

- true - the workload manager attempts to satisfy failover using any available instance.
- false - failover does not occur.

Related Information

Failover Mode [page 106]
fail_to_any Attribute [page 106]

12.6.1 Adding Failover Resources

Add failover resources to a logical cluster using `sp_cluster logical, "add"`. You cannot add failover resources to the system logical cluster.

Each time you use `sp_cluster logical, "add"` to add failover resources, SAP ASE Custer Edition creates one or more failover groups.

If you add one or more failover instances, separating multiple instances with semicolons, SAP ASE Cluster Edition places all instances in a single group.

This example adds "ase3" as a failover group to "SalesLC":

```
sp_cluster logical, "add", SalesLC, failover, ase3
```

You can also add failover instances to existing failover groups. For example, if "ase3" is a member of failover group 1, to add "ase4" to failover group 1, enter:

```
sp_cluster logical, "add", SalesLC, failover, ase4, "1"
```

View failover resource information, including the failover group ID, using `sp_cluster logical, "show"`.

12.7 Managing Logical Clusters

Managing logical clusters involves tasks such as creating and dropping a logical cluster, adding and dropping resources, and migrating connections.

12.7.1 User Tasks and Logical Clusters

Each SAP ASE Cluster Edition task (identified by a server process ID) runs inside a logical cluster. The `lcid` column in `sysprocesses` is a logical cluster ID that is hosting a given task. This ID can be passed to the `lc_name()` built-in function to determine the name of the corresponding logical cluster.

An individual task may run the `lc_name()` function to determine the current logical cluster.

12.7.2 Workload Manager Thread

The workload manager thread is a system-service thread that runs on each instance.

When an instance starts, it automatically spawns the workload manager thread. This thread spends most of its time sleeping, but wakes up periodically for management duties such as handling a logical cluster action, gathering workload metrics, calculating the load on each instance, and sending load information to all instances.

View information about the workload manager by querying `sysprocesses` and `monProcesses`, using `sp_who`, and using other SAP ASE Cluster Edition capabilities for monitoring processes.

You may want to change the default value for maximum memory usage for the workload manager. Otherwise, the workload manager requires no maintenance.

12.7.2.1 Setting Memory Requirements for the Workload Manager

Set the `workload manager cache size` configuration parameter to specify the maximum amount of memory that the workload manager can use. All memory used by the workload manager comes from the memory pool sized by `workload manager cache size`.

Connection migration consumes memory from this pool; each configured logical cluster, route, and load profile consumes memory from this pool. Actions resulting from commands such as `failover` and `failback` consume memory from this pool and continue to do so until the action is released.

Estimate the memory usage based on these guidelines:

- Four memory pages for each concurrent migrating connection
- One page for three logical clusters

- One page for two load profiles
- One page for 30 routes
- One page for 12 actions

Set the maximum value of the memory pool in increments of 2K pages using `sp_configure`.

This example sets the value of `workload manager cache size` to 100 2K pages:

```
sp_configure "workload manager cache size", 100
```

> **Note**
>
> `workload_manager_size` is dynamic; you need not restart the server. The default value is 80, or 160KB.
>
> The default value is sufficient for most installations. If you anticipate migrating a logical cluster with many concurrent connections, you may need to increase the size of the memory pool.

12.7.3 Viewing Information about a Logical Cluster

View information about a logical cluster using built-in functions, global variables. monitoring tables, or `sp_cluster logical, show`.

- Use the built-in functions `lc_name()`, `lc_id()`, `instance_name()`, and `instance_id()`.
- Use the global variables `<@@clustername>`, `<@@instancename>`, and `<@@instanceid>`.
- Query the monitoring tables.
- Use `sp_cluster logical, show`.

12.7.3.1 Related Monitoring Tables

Some monitoring tables provide information about a logical cluster, the workload, and the workload profile.

- `monLogicalCluster` – provides summary information about the system's logical clusters.
- `monLogicalClusterInstance` – provides information about each instance in the system's logical clusters.
- `monLogicalClusterRoute` – provides information about configured routes.
- `monLogicalClusterAction` – provides information about actions in the system's logical clusters.
- `monWorkload` – provides the workload scores for each instance for each load profile.
- `monWorkloadProfile` – provides information about each load profile.

Use Transact-SQL commands to query these tables for information. See the SAP ASE *Reference Manual: Tables* for a complete description of each of the monitoring tables.

12.7.3.2 Using sp_cluster logical,"show"

Use `sp_cluster logical, "show"` for multiple purposes.

- To view summary information about a specific logical cluster, use:

  ```
  sp_cluster logical, "show", <lc_name>
  ```

  This example displays information about "SalesLC":

  ```
  sp_cluster logical, "show", SalesLC
  ```

  To view information about all logical clusters, enter:

  ```
  sp_cluster logical, "show"
  ```

- To view information about a administrative actions, use the following, where the value of `<state>` is cancelled, complete, or active:

  ```
  sp_cluster logical, "show", <lc_name>, action[, <state>]
  ```

  This example displays information about all completed actions:

  ```
  sp_cluster logical, "show", NULL, action, complete
  ```

  This example displays information about all completed actions for "SalesLC":

  ```
  sp_cluster logical, "show", SalesLC, action, cancelled
  ```

  This example displays information about active actions for "Sales LC":

  ```
  sp_cluster logical, "show", SalesLC, action, active
  ```

- To view information about configured routes, use the following, where the value of `<type>` is application, alias, or login:

  ```
  sp_cluster logical, "show"[, <lc_name>, route[, <type>]
  ```

  You can query on a particular application, login, alias or combination of these.
  This example displays information about the "sales_web_user" login route to "SalesLC":

  ```
  sp_cluster logical, "show", SalesLC, route, login, sales_web_user
  ```

For complete syntax and usage information for `sp_cluster logical, "show"`, see the *SAP ASE Reference Manual: Procedures*.

12.7.4 Creating and Dropping a Logical Cluster

Create and drop logical clusters using the `sp_cluster logical` system procedure.

- Create a logical cluster using `sp_cluster logical, "create"`.
  This example creates "FinanceLC":

  ```
  sp_cluster logical, "create", FinanceLC
  ```

- Drop a logical cluster using `sp_cluster logical, "drop"`. A logical cluster must be offline or inactive before it can be dropped. `sp_cluster logical, "drop"` deletes the cluster and all routes, resources, and attributes associated with the cluster.
  This example drops "FinanceLC":

  ```
  sp_cluster logical, "drop", FinanceLC, cluster
  ```

12.7.5 Adding Resources to a Logical Cluster

Add resources to a logical cluster using the `sp_cluster logical, "add"` system procedure

You can add:

- Base instances
- Failover instances

Related Information

Adding Instances to a Logical Cluster [page 100]
Adding Failover Resources [page 110]

12.7.6 Dropping Resources from a Logical Cluster

Drop one or more resources from a logical cluster using the `sp_cluster logical, "drop"` system procedure. A base instance or a failover instance must be offline before it can be dropped.

You can drop:

- Base instances – this example drops instance "ase2" from "SalesLC":

  ```
  sp_cluster logical, "drop", SalesLC, instance, ase2
  ```

- Failover instances – this example drops failover instances "ase3" and "ase4" from "SalesLC":

  ```
  sp_cluster logical, "drop", SalesLC, failover, "ase3;ase4"
  ```

12.7.7 Adding, Moving, and Dropping Routes

Add, move, and drop routes using the `sp_cluster logical` system procedure.

- Add a route using `sp_cluster logical, "add"`.
  This example adds a route for the logins "accounting" and "projects" to "SalesLC":

  ```
  sp_cluster logical, "add", SalesLC, route, login, "accounting;projects"
  ```

- Move a route from one logical cluster to another using `sp_cluster logical, "alter"`.
  This example creates a route to "My_LC" using the alias "SalesLC", and then move the route from "My_LC" to "Your_LC":

  ```
  sp_cluster logical, "add", My_LC, route, alias, SalesLC
  sp_cluster logical, "alter", Your_LC, route, alias, SalesLC
  ```

- Drop a route using `sp_cluster logical, "drop"`.
  This example drops the login "projects" from "SalesLC":

  ```
  sp_cluster logical, "drop", SalesLC, route, login, projects
  ```

12.7.8 Migrating Connections

Migrate conections using the `sp_cluster` system procedure.

- Use `sp_cluster connection "migrate"` to:
  - Migrate the logical cluster or instance on which a connection (or another task) is running.
  - Migrate an application or login to a logical cluster or instance for which it is not routed.

  This example migrates the connection with spid 73 to the "SalesLC" logical cluster:

  ```
  sp_cluster connection, "migrate", SalesLC, NULL, "73"
  ```

  See the *SAP ASE Reference Manual: Procedures*.
- Use `sp_cluster logical` to set a manual or automatic migration to another logical cluster, or gather groups of connections to another logical cluster when predefined events occur. You can gather all qualified connections on the system or logical cluster to a designated logical cluster using the defined routing rules. The SAP ASE Cluster Edition looks for all connections that match the routing rules for this logical cluster, and migrates them to the specified logical cluster.
  The syntax is:

  ```
  sp_cluster logical, 'gather', <lc_name>
  sp_cluster logical 'set', <lc_name>, 'gather', 'automatic | manual'
  ```

  This example gathers all the defined connections to the "SalesLC" logical cluster:

  ```
  sp_cluster logical, 'gather', SalesLC
  ```

  This example sets the gathering to `manual` for the "SalesLC" logical cluster:

  ```
  sp_cluster logical 'set', SalesLC, 'gather' 'manual'
  ```

  See the *SAP ASE Reference Manual: Procedures*.

12.8 Administering Logical Cluster State

Manually change the state of a logical cluster and its instances using `sp_cluster logical, <action>`.

Valid values for `<action>` are:

- `failover`
- `failback`
- `online`
- `offline`
- `deactivate`

12.8.1 Failover

Specifying the `failover` parameter initiates a manual failover from a logical cluster's base resources to its failover resources.

To use the failover feature, set up your failover resources using `sp_cluster logical, "add"`. When initiating a partial-cluster failover, specify a list of base resources that are to fail over, and a list of failover resource to which the base instances will fail over.

For example, you can fail over a portion of a logical cluster to a set of previously configured failover resources. In this example, "SalesLC" is running on instances "ase1" and "ase2". To keep "SalesLC" running on "ase2", but fail over "ase1" to the previously defined failover resource "ase3", enter:

```
sp_cluster logical, "failover", SalesLC, instance, ase1, ase3
```

12.8.2 Failback

The `failback` parameter reverses a failover.

Specifying the `failback` parameter initiates a manual failback from a logical cluster's failover resources to its base resources. When initiating a partial-cluster failover, you specify a list of failover resources that are to fail back, and a list of base resources to which the failover instances will fail back.

This example incrementally fails back "SalesLC", which is running on "ase3", so that "SalesLC" runs on "ase1" and specifies a 2-minute wait.

```
declare @out_handle varchar(15)
```

```
execute
sp_cluster logical, "failback", SalesLC, instance, ase3, ase1, wait, "00:02:00",
@handle = @out_handle output
```

12.8.3 Online

The `online` parameter starts a logical cluster or instances in a logical cluster, placing them in the online state.

This example starts "SalesLC" on "ase1":

```
sp_cluster logical, "online", SalesLC, ase1
```

Related Information

Starting a Logical Cluster [page 102]

12.8.4 Offline

The `offline` parameter stops logical clusters or instances that are in the online or active state.

This example takes "SalesLC" offline, wait 5 minutes, then stores the action in an action handle in a local variable:

```
declare @out_handle varchar(15)
```

```
execute
sp_cluster logical, "offline", SalesLC, cluster, wait, 00:05:00,
@handle=@out_handle output
```

12.8.5 Deactivate

The `deactivate` parameter is identical to `offline`, except that it puts the cluster or instance in the inactive state.

Related Information

Offline [page 117]

12.8.6 Cluster and Instance States

A logical cluster and each instance in the cluster can have different states.

- A logical cluster has an overall, or global, state that determines, for example, whether the cluster is offline or online.
- A logical cluster also has an instance state that describes the state of a particular instance as it is perceived by a logical cluster. For example, an online logical cluster may be online on its base instances and offline on its failover instances. This state may be independent of the actual SAP ASE state, as a logical cluster may be offline on an instance that is actually up and running.

Table 3: Logical Cluster States

State	Global Level	Instance Level
online	A logical cluster is online and running on one or more instances.	The online logical cluster is accepting and managing connections on the current instance.
offline	A logical cluster is not running on any instance.	A logical cluster is not running on the current instance, which cannot accept connections or consume resources.
inactive	Similarly to the offline state, a logical cluster is not running on any instance. Inactive logical clusters are not started automatically and do not participate in failover. The cluster achieves the inactive state only through the `deactivate` command. Once inactive, the cluster comes online only through the `online` command.	A logical cluster is not running on the current instance, which cannot accept connections or consume resources. In addition, the inactive instance cannot fail over and is not brought online after an automatic start-up. This state is achieved only through the `deactivate` command.
failed	Similarly to the offline state, a logical cluster is not running on any instance. A logical cluster moves to the failed state when its active instances are shut down using `shutdown with nowait` command or encounter a system failure when no failure resources are available.	A logical cluster is not running on the current instance, which cannot accept connections or consume resources. The failed state is reached through `shutdown with nowait` command or system failure.
time_wait	A transition state between online and offline or inactive. An online logical cluster must enter the `time_wait` state before becoming offline or inactive. During this time, any new connections are routed according to the down-routing mode, and existing connections migrate or disconnect.	A transition state between online and offline or inactive. When a logical cluster is online for an instance, it must enter the `time_wait` state before becoming offline or inactive. In the `time_wait` state, no new connections can be routed to a logical cluster or instance, but existing connections continue to run until they migrate or disconnect.

12.8.7 How States Change

Cluster and instance states can change manually or automatically.

- Manually, when you execute a state change using the `online`, `offline`, `failover`, and `failback` commands; and sometimes with the `action` command.
- Automatically, as a result of system changes.

The initial state of the cluster or instance can dictate whether or not a state change is valid, and even the final state.

Table 4: Interaction of Action and State

	offline	online	time_wait	failed	inactive
Online	`online`			`online`	`online`
Offline		`offline/ time_wait`		`offline`	`offline`
Failback instance	`online`			`online`	`online`
Failback cluster		`online/ time_wait`		`offline`	`offline`
Failover instance	`online`			`online`	`online`
Failover cluster		`online/ time_wait`		`offline`	`offline`
Cancel action			`online`		
Modify wait			`time_wait`		
Deactivate	`inactive`	`inactive/ time_wait`		`inactive`	

States can also change as the result of system changes. The following table shows the effects of different system changes on the state of the cluster or instance.

Table 5: System Changes on the State of the Cluster or Instance

	offline	online	time_wait	failed	inactive
Instance joins cluster	`online` if automatic start-up is configured			`online` if automatic start-up is enabled	
Graceful shutdown		`time_wait`			
System failure		failed	failed		

	offline	online	time_wait	failed	inactive
Shutdown with nowait		failed	failed		
Failover selection	online				

> **Note**
>
> Logical cluster states are not retained following a total cluster restart. For example, if you execute the `offline` command for a logical cluster that is in automatic start-up mode, the cluster is in the online state after you restart it.

12.8.8 Asynchronous Commands and Logical Cluster States

The `sp_cluster logical` commands `deactivate`, `failback`, `failover`, and `offline` are asynchronous. They stop an online instance that may have existing transactions.

Since you perform these existing transactions before you take an instance offline or make it inactive, you can complete these commands at a later time and context.

When you execute any of these commands, the target instance is placed in the `time_wait` state, and no longer accepts new connections.

Each asynchronous command provides three "wait" options for handling existing transactions:

- `wait` – lets existing connections remain for a specified period of time, for example five minutes. Connections that support migration migrate as soon as they are quiescent. Connections that do not support migration disconnect when they become quiescent. HA-aware clients fail over; clients that are not HA-aware disconnect. Connections remaining after the specified time are terminated.
- `until` – lets existing connections remain until a specified time, for example 12:30 p.m. Otherwise, `until` and `wait` handle connections in the same way.
- `nowait` – terminates existing connections immediately. Connections that support migration must migrate immediately, or they are terminated.

> **Note**
>
> If you do not specify a wait option when you execute an `sp_cluster logical` asynchronous command, SAP ASE Cluster Edition assumes an infinite wait.

When the last connection using the instance disconnects, the instance state changes from `time_wait` to `offline` or `inactive`.

12.8.9 Using Action Descriptors

Track or change an action using action descriptors.

An asynchronous command generates an action descriptor when it seeks to stop one or more instances. The action descriptor tracks the action, the wait option, and target instances in the `time_wait` state. You can view information about an action descriptor by querying the `monLogicalCLusterAction` table or executing `sp_cluster logical, "show", NULL, action.`

An action can be "active" or "complete".

- An action is active when at least one target instance remains in the `time_wait` state.
- An action is complete when all target instances are no longer in the `time_wait` state.

Manage action descriptors using `sp_cluster logical, action` and these options:

- `cancel` – terminates active actions. Instances in the `time_wait` state due to that action return to the online state. Existing connections remain, even if they were marked for migration or termination.
  If the command that resulted in the creation of the action brought instances online, they remain online. For example, if an action results in the cancellation of a failover from s1 to f1, f1 remains online.
- `modify_wait` – changes the wait option and time associated with an active action. For example, if an action is created with a 10-minute wait, use `modify_wait` to change:
  - The time delay to 20 minutes
  - The time delay to the actual clock time of 4:30 p.m.
  - The wait option to `nowait`
- `release` – removes a completed action from the `monLogicalClusterAction` table. Completed actions remain in the `monLogicalClusterAction` table so you can track their status. However, completed actions consume memory from the workload manager cache. Execute the `release` command after an action completes to free this memory.

> **Note**
>
> Action information is stored in memory only. Restarting the full cluster clears all actions from the `monLogicalClusterAction` table.

Related Information

Asynchronous Commands and Logical Cluster States [page 120]

12.8.10 Scheduling and Rescheduling a Failover

Execute an administrative failover for a cluster or an instance. The cluster or instance fails over to previously configured failover resources.

This example fails over the "SalesLC" cluster, scheduling the failover for 2 a.m. In order to later track or change the action, syntax is included that generates an output for an action handle:

```
declare @out_handle varchar(15)
execute
sp_cluster logical, "failover", SalesLC, cluster, NULL, until, "02:00:00",
@handle = @out_handle output
```

If the command outputs the action handle "1234", and "SalesLC" enters the `time_wait` state, all new connections migrate to the failover resources. Any existing connections remaining after 2 a.m. are terminated, and "SalesLC" enters the offline state.

If all connections must migrate immediately, use the action handle to reschedule an immediate failover:

```
sp_cluster logical, "failover", SalesLC, modify_time, "1234", nowait
```

12.9 Workload Distribution

Each instance has a workload manager thread that is responsible for calculating the load on the current instance and sending that information to the other instances in the cluster. The workload manager is a system-service thread; it is spawned when the instance starts.

SAP ASE Cluster Edition uses a workload measurement algorithm to calculate a load score for each instance. This load score is a unitless number that can be compared with other load scores to determine relative workloads. Thus, you can compare load scores across a cluster or at different times for a particular cluster. A load score is meaningful only when compared to other load scores.

12.9.1 Workload Metrics

The workload manager recalculates load scores every 15 seconds, using the load scores (and the resulting statistics) to compare the relative workloads for all instances in the cluster.

The workload manager uses the statistics to populate the monitoring tables. The statistics are per-instance: the workload manager does not track per-spid statistics.

When calculating a load score, SAP ASE Cluster Edition considers five system-defined metrics and, optionally, one user-supplied metric:

- User connections – the capacity of an instance to accept a new connection, based on resource availability.
- CPU utilization – the capacity of an instance to accept additional work.
- Run-queue length – the number of runnable tasks on a system. Run-queue length measures the processing backlog, and is a good indicator of relative response time.

- I/O load – outstanding asynchronous I/Os.
- Engine deficit – the difference in the number of online engines among instances in the cluster.

> **Note**
>
> Engine deficit is measurable only when instances in the cluster have unequal numbers of engines. In this scenario, engine deficit adds a metric for maximum relative capacity to the load score.

- User metric – an optional metric specific to the user's environment. Use the `workload_metric` built-in function to supply its value.

The load score is calculated with this formula:

```
ConnectionsWeight  x  (ConnectionsLoad)/100
+                        CPUWeight x (CPULoad)/100
+      RunQueueWeight x (RunQueueLoad)/100
+ UserDefinedWeight x (UserDefinedLoad)/100
+                  EngineWeight x (EngineLoad)/100
+                                IOWeight x (IOLoad)/100
-------------------------------------------------------------
=                                                   Load score
```

Related Information

Creating a User Metric [page 123]

12.9.2 Updating a User Metric

With the `workload_metric` built-in function, you can update the value of the user metric for an instance, which is used to calculate the total load score of an instance. You can include `workload_metric` in a user-defined stored procedure, trigger, or an externally triggered script. The instances recalculate the load score every 15 seconds.

12.9.3 Creating a User Metric

Add a site-specific metric to the workload measurement algorithm using the `workload_metric` built-in function.

In a typical example, you might monitor response time using an external monitor, then insert the response-time values into the algorithm. SAP ASE Cluster Edition normalizes system-supplied metrics before including them in the workload algorithm. For compatibility, normalize the user-supplied metric as well. For example, if the maximum acceptable response time is 5 seconds and the measured response time is 2 seconds, the normalized value is 40 (2 is 40% of 5); enter this into the workload algorithm using `workload_metric`.

12.9.4 Weighting the Workload Metrics

Using SAP Control Center or `sp_cluster`, you can enter and adjust the weighting factors, which are visible to the system administer to customize the Workload Profiles.

Each component of the load score is weighted according to importance. The metric values are normalized, and the results are summed to give each instance an overall load score. Default values are sufficient for most sites, but if you are including a site-specific metric, you may want to adjust the weights using `sp_cluster profile`.

12.9.5 Load Thresholds

SAP ASE Cluster Edition uses the load score for each instance to determine how best to distribute incoming connections (login redirection) and whether to migrate existing connections (dynamic load distribution).

Load distribution is performed only for a logical cluster running on multiple instances. The SAP ASE Cluster Edition load distribution redirects work when one instance is overloaded and other instances are available. It does not seek to maintain perfectly balanced load scores. As a result, SAP ASE Cluster Edition maintains load thresholds for login redirection and dynamic load distribution. The load threshold is the percentage difference between the load on the current instance and the load on the least loaded instance currently participating in a logical cluster. That value must be met before SAP ASE redirects a login or migrates an existing connection.

SAP ASE Cluster Edition maintains separate load thresholds for login redirection and connection migration. Typically, the login redirection threshold is lower than that for connection migration. Use `sp_cluster profile` to configure the load thresholds when creating a load profile.

Hysteresis Value

The load threshold tells SAP ASE Cluster Edition when to redirect connections from an overloaded instance to the least loaded instance. The hysteresis value guards against migration when the threshold value is met, but the actual load scores are too low to require migration.

For example, suppose the load score on the current instance is 2 and that of the least loaded instance is 1. The percentage difference is 100%, which meets the load threshold, but the actual load scores are so low that migration is not appropriate.

12.9.6 Load Profiles

Load profiles consolidate all configurable aspects of the workload scoring system into a single named entity.

You can assign different load profiles to different logical clusters in the same physical cluster, thus enabling multiple applications with diverse workloads within the same physical cluster. You can also share the same load profile among multiple logical clusters.

For example, a shared-disk cluster can simultaneously host a DSS-based logical cluster that is primarily read-only, and an OLTP logical cluster that handles frequent writes. The optimum thresholds for login redirection

and connection migration for these two clusters may differ greatly. The ability to assign specific load profiles to each logical cluster allows each cluster to function more efficiently.

> **Note**
>
> SAP ASE Cluster Edition gathers load statistics for each instance without regard to logical cluster. Thus, if two logical clusters run on the same instance, they have the same raw data for that instance. However, each logical cluster interprets and uses that data according to its own load profile.

You can use the preconfigured profile created for an OLTP environment. You can also create your own load profiles using `sp_cluster profile`.

12.9.7 Login Redirection

Login redirection allows an instance in a logical cluster to redirect an incoming client connection to another instance.

For example, a connection from a field sales application bound to the "SalesLC" logical cluster may be redirected from a heavily loaded instance to the least loaded instance currently hosting "SalesLC". The logic for making the redirection decision in based on routing rules and workload data.

You can specify whether or not client connections may be redirected. By default, redirection is enabled. To disable redirection, `call ct_conn_props()`, `ct_config`, or edit the `ocs.cfg` file.

When a client attempts to connect to the logical cluster, SAP ASE Cluster Edition determines whether or not redirection is enabled. If yes, SAP ASE Cluster Edition checks routing rules that specify the logical cluster, then returns a list of available instances in that cluster, ordered from least to most heavily loaded.

12.9.8 Connection Migration

Connection migration enables an instance in a logical cluster to migrate an existing connection to another instance.

Only connections that are transactionally quiescent can migrate. That is, migration can take place:

- Upon receipt of a batch from a client
- Upon completion of a batch
- While waiting for a batch (totally idle)

Connection migration may occur automatically when a task determines that it should migrate to a new instance based on the current workload within the cluster. Usually connection migration occurs as a result of one of the following `sp_cluster logical` system procedure options issued by the system administrator:

- `deactivate`
- `failback`
- `failover`
- `migrate`
- `offline`

Graceful migration from one instance to another is usually an asynchronous event. Typically, the system administrator issues a `migrate` or other command with a timeout before the migration is to occur. When the migration command is issued, all tasks running in the logical cluster are flagged to migrate. As each task crosses a migration point (receipt or completion of a batch), it checks to see if it can migrate. If not, the task continues to check at each migration point until it is eligible for migration, and migrates, or the timeout expires.

In this example, a system administrator plans to take instance "Server1" in "SalesLC "offline at 5pm. The administrator could trigger a migration at 4:30 with a 30-minute timeout.

```
sp_cluster logical, "offline", "SalesLC', "instance", "server1", "wait",
"00:30:00"
```

Eligible connections would move to new instances immediately. The 30-minute timeout should allow enough time for active connections to reach a state with no open transactions or active commands. At the end of the timeout, any remaining connections terminate. If the client supports HA, terminated connections may trigger an HA-failover to a new instance.

12.10 Sample Load Profiles

SAP ASE Cluster Edition includes two sample load profiles named `sybase_profile_oltp` and `sybase_profile_dss`.

- `sybase_profile_oltp` – configured for OLTP environments. It tries to keep all connections on the same instance by disabling load-based login distribution and dynamic-load distribution. Emphasis is placed on Run-queue depth, which is a good predictor of response time.
- `sybase_profile_dss` – configured for primarily read-only, DSS environments. It uses load-based login distribution and dynamic load distribution to distribute load across multiple instances, but places an emphasis on balancing CPU usage and user connections.

This table lists the metrics for `sybase_profile_oltp` and `sybase_profile_dss`.

Table 6: Metrics for the Sample Load Profiles

Workload Metric	sybase_profile_oltp	sybase_profile_dss
Profile ID	1	2
User connections	0	40
CPU utilization	10	40
Run-queue length	70	0
I/O load	20	10
Engine deficit	0	10
User weight (not used)	0	0

Workload Metric	sybase_profile_oltp	sybase_profile_dss
Login threshold	0	20
Dynamic threshold	0	50
Hysteresis	20	20

12.11 Creating and Configuring Your Own Load Profile

Create and configure your own load profile.

Procedure

1. Create the empty load profile.
2. Build the load profile by specifying individual metric weights and thresholds.
3. Associate the load profile with a logical cluster.

12.11.1 Creating the Load Profile

Build the empty load profile using `sp_cluster profile, "create"`.

This example creates the profile "my_profile":

```
sp_cluster profile, "create", my_profile
```

12.11.2 Building the Load Profile

Build the load profile by specifying a weight for each of the metrics that make up the load profile and load distribution thresholds.

12.11.2.1 Specifying Weights for Load Profile Metrics

Use `sp_cluster profile, "set"` to configure weights for each of the metrics covered in the load profile.

The attributes are:

- User connections
- CPU busy
- Run-queue length
- I/O load
- Engine deficit
- User metric

Set each metric individually by using values between 0 and 255. This example sets weights for "my_profile":

```
sp_cluster profile, "set", my_profile, weight, "user connections", "0"
sp_cluster profile, "set", my_profile, weight, "cpu busy", "20"
sp_cluster profile, "set", my_profile, weight, "run queue", "30"
sp_cluster profile, "set", my_profile, weight, "io load", "10"
sp_cluster profile, "set", my_profile, weight, "engine deficit", "10"
sp_cluster profile, "set", my_profile, weight, "user metric", "30"
```

Related Information

Creating a User Metric [page 123]
Workload Metrics [page 122]

12.11.2.2 Specifying Load Distribution Thresholds

Set load distribution thresholds values between 0 and 100 using `sp_cluster profile, "set"`.

A value of zero (0) disables that aspect of load distribution. Set separate load thresholds for:

- Login redirection
- Dynamic load distribution
- The hysteresis value

This example turns off dynamic load distribution in "my_profile":

```
sp_cluster profile, "set", my_profile, threshold, "dynamic", "0"
```

This example sets the login redirection threshold to 30 for "my_profile":

```
sp_cluster profile, "set", my_profile, threshold, "login", "30"
```

This example sets the hysteresis value to 20 for "my_profile":

```
sp_cluster profile, "set", my_profile, threshold, "hysteresis", "20"
```

12.11.3 Associating the Load Profile with a Logical Cluster

Associate a load profile with a logical cluster using `sp_cluster logical, "set"`.

This example associates the profile "my_profile" with "SalesLC":

```
sp_cluster logical, "set", SalesLC, load_profile, my_profile
```

12.11.4 Changing the Load Profile

Each logical cluster is associated with either a system load profile or a user-created load profile. To change the load profile, associate the new load profile with a logical cluster.

This changes the load profile for "SalesLC" from "my_profile" to "sybase_profile_oltp":

```
sp_cluster logical, "set", SalesLC, load_profile,
sybase_profile_oltp
```

You can then drop the old load profile. This example drops "my_profile":

```
sp_cluster profile, "drop", my_profile
```

12.12 Troubleshooting

There are several trace flags that you can use to troubleshoot the workload manager.

Table 7: Trace Flags for the Workload Manager

Trace Flag	Traces
16403	SPID through connection to and disconnection from a logical cluster.
16404	Routes and route resolution.
16406	Client redirection and migration.

Trace Flag	Traces
16414	Changes in a logical cluster state machines, including transitions from online to offline, and so on.

13 I/O Fencing

With I/O fencing, a membership manager builds a fence around the data storage, and allows only instances or nodes that behave properly to perform writes. Use I/O fencing to ensure data integrity, and to prevent data corruption from uncooperative cluster members.

A shared-disk cluster can detect the presence of a noncooperating instance and remove it from the cluster. In rare situations, however, the cluster cannot stop the noncooperating instance from writing to the shared disk, even though the instance is no longer part of the cluster. For example, if an instance has been removed from the cluster, but has not released resources and shut down, it may still be able to write to the shared disk. Enabling I/O fencing prevents the noncooperating instance from making data writes.

> **Caution**
>
> Only test and development environments that can accept this risk should be deployed without I/O fencing.

SAP ASE Cluster Edition supports the SCSI-3 Persistent Group Reservation (PGR) feature of SCSI-3 devices to provide I/O fencing. PGR is the SCSI-3 standard for managing disk access in an environment where a single disk is shared by multiple hosts for read and write access.

The I/O fencing provided by the SCSI-3 PGR feature operates only on devices, not on partitions. For example, `/dev/sda1` and `/dev/sda2` are partitions of the device `/dev/sda`. A fencing operation targeted to a raw device bound to `/dev/sda1` affects all partitions of `/dev/sda`, so any file systems or other applications (including another SAP ASE) using partitions on that device are also affected. For this reason, the device must be used exclusively by the cluster instance.

> **Note**
>
> SCSI-3 PGR functionality is available only for a physical SCSI disk device or a virtual disk device exported by a storage area network (SAN). Partitioning such a device at the operating system level does not provide SCSI-3 PGR on each partition. Rather, the SCSI-3 PGR, and thus the fencing support, is shared among all partitions on the device. SAP ASE cannot fence database devices independently. Do not create database devices on partitions that use the same device, as they cannot be fenced independently. The quorum must reside on its own device. SAP ASE does not allow you to create database devices on the quorum device. In addition, if device partitions are used outside the cluster, any fencing performed by the cluster denies access to those other partitions for the external applications or file systems placed on them.

I/O fencing is based on a device driver API that is targeted to a specific device driver. The device driver is typically called the fence device. Use operating system commands to create the fence device on each of the nodes running the cluster.

13.1 Setting Up I/O Fencing

Enable I/O fencing by turning on the `enable i/o fencing` parameter.

To enable I/O fencing:

- Each instance in the cluster must run on a separate node.
- All storage devices (disks) that hold database devices must support the SCSI-3 standard and cannot be partitioned.

To turn on the `enable i/o fencing` configuration parameter, enter:

```
sp_configure "enable i/o fencing", 1
```

The SCSI-3 PGR feature is platform-dependent, and all devices used by the SAP ASE Cluster Edition should have this functionality. See your operating system documentation for complete syntax and setup information.

As part of the cluster creation process, both the SAP Control Center and `sybcluster` let you check that each device is enabled for I/O fencing. You can also run the `qrmutil` utility:

```
$SYBASE/$SYBASE_ASE/bin/qrmutil
```

> **Note**
>
> Configuring I/O fencing affects all devices with I/O fencing capability system-wide.

13.1.1 Setting Up I/O Fencing on Linux

Use operating system commands to load a SCSI generic (sg), driver on each node running the shared-disk cluster. Loading the driver automatically creates `/dev/sg*` files.

Grant read and write permission to each instance for all `/dev/sg*` files created on that node by the sg driver.

For example, if the database device on the raw partition `/dev/raw/raw1` is mapped to `/dev/sg3`, grant write permission on the `/dev/sg3` file.

Make sure that the SAP ASE Cluster Edition has write access to `/dev/sg` files that correspond to configured database devices. In addition, the fence device must have write access to `/dev/sg` files containing SCSI-3 PGR commands.

The path for the fence device may differ from the corresponding physical device path. For example, the database device might be `/dev/raw/raw1`, the physical device `/dev/sda`, and the fence device `/dev/sg0`.

> **Note**
>
> Whenever the sg driver is loaded at system startup (or through user command), the system re-creates the `/dev/sg*` files. If you do not make sure that permissions are granted appropriately on these files as part of the startup process, the system administrator must manually reset permissions each time the system starts.

14 Quorum Device Heartbeat

Periodically, each instance in the cluster "checks in" with the quorum device, creating a quorum device heartbeat that enables instances to monitor the status of the cluster.

Instances use the device heartbeat to:

- Determine whether the quorum device is accessible. If an instance fails to write the heartbeat to the quorum device, the instance may have has lost its storage area network (SAN) link, or it may be blocked from the cluster devices.
- Read the heartbeat values from the quorum disk while it is starting. If the instance detects no changes in the heartbeat after a configured period of time, the instance determines the cluster is not running.

Instances do not use the quorum heartbeat to detect instance failure.

14.1 Configuring the Quorum Device Heartbeat

You can configure how often the heartbeat happens and the number or times an instance attempts to detect the quorum heartbeat before it assumes the quorum device is not functioning with the `quorum heartbeat interval` and the `quorum heartbeat retries` configuration parameters, respectively.

quorum heartbeat interval

The `quorum heartbeat interval` configuration parameter specifies the number of seconds between quorum heartbeats. The default is 5, so each instance writes a quorum heartbeat once every 5 seconds. The minimum value is 1 second and the maximum value is 60 seconds.

Most sites need not tune this configuration parameter. Setting `quorum heartbeat interval` to a lower number increases the heartbeat overhead, but speeds the detection of a lost disk link, so that instances terminate more quickly if they are blocked or have lost their SAN link. Setting `quorum heartbeat interval` to a higher value reduces the heartbeat overhead, but delays the detection of a lost disk link. The amount of overhead caused by the heartbeat depends on the performance of your disk subsystem.

quorum heartbeat retries

The `quorum heartbeat retires` configuration parameter specifies the number of times an instance attempts to detect a quorum heartbeat before determining that the quorum device is no longer running, then exiting. The default is 2, which means that the instance terminates after the third consecutive quorum heartbeat failure because the first two failed. The minimum value is 0, which indicates that the instance should terminate upon the first quorum heartbeat failure. The maximum value is 32,768.

Tuning this configuration parameter to a lower number causes an instance to fail over more quickly when access to the quorum device is lost, potentially improving application recovery times. Tuning it to a higher number degrades application recovery, reducing the chances that a transient disk access problem causes an instance failure.

15 Distributed Checkpoints

Checkpoints control recovery in both nonclustered SAP ASE and SAP ASE Cluster Edition systems, but the process for which they supply the transaction information differs.

In the SAP ASE Cluster Edition, the oldest active transaction found is the oldest active transaction across the entire cluster, and dirty buffers are flushed on all clustered instances.

In both types of processing, SAP ASE finds the oldest active transaction information at the time the checkpoint record is logged, uses it as a cut-off point, and writes the checkpoint record to the log. The dirty buffers are then flushed, up to the current flush sequence number. This checkpoint record is registered as a recovery checkpoint only after all the dirty buffers are flushed.

15.1 Cluster Checkpoints

In a distributed cluster checkpoint, flush sequence numbers are local to individual instances. The instance that initiates the checkpoint is the initiator.

The initiator coordinates the checkpoint process among all instances. This means sending checkpoint requests to the other instances and receiving their answers. All instances flush dirty buffers upon receiving the checkpoint request from the initiating instance. Messages among the initiating instance and other instances are posted to a dedicated `cipc` endpoint.

15.1.1 Execution of a Distributed Checkpoint

The process of executing a distributed checkpoint includes multiple steps.

1. The master instance writes a new checkpoint record, if needed.
2. The master instance inserts a checkpoint request in the local list of checkpoint requests.
3. The master instance flushes its own dirty buffers.
4. The other instances generate a new checkpoint task upon receiving the request from the master instance.
5. The new checkpoint task gets the current local sequence number in the database, flushes local dirty buffers that have a flush sequence number that is smaller than the current sequence number, and returns a reply to the initiating instance.
6. The master instance remains in a loop, waiting for all replies from other instances. After each reply, the master removes the sender instance from the instance map. When the instance map is empty, the master instance registers the checkpoint record in `dbinfo`.

16 Distributed Transactions on a Shared-Disk Cluster

SAP ASE supports distributed transaction management (DTM) on its clustered architecture.

DTM on the SAP ASE Cluster Edition:

- Is fully compliant with the X/Open XA protocol when it acts as the resource manager (RM), without additional services, such as XA-Server.
- Ensures consistent commit or rollback for transactions that update SAP ASE data via remote procedure calls (RPCs) and Component Integration Services (CIS).
- Can be part of distributed transactions coordinated by other SAP ASE installations using the SAP ASE Transaction Coordination (ASTC) mechanism.
- Can coordinate distributed transactions across multiple SAP ASE installations using the ASTC mechanism.

> **Note**
>
> The SAP ASE Cluster Edition does not support the Microsoft Distributed Transaction Coordinator (MSDTC) proprietary protocol.

16.1 Using DTM on the Shared-Disk Cluster

In general, the user interface for distributed transactions is the same in the SAP ASE Cluster Edition as in the nonclustered environment, but there are some cluster-specific issues exist.

Applications using DTM on a nonclustered SAP ASE can use the same applications on a clustered SAP ASE. See the *Using SAP ASE Distributed Transaction Management Features*.

16.2 The Cluster as the Resource Manager

Although the cluster comprises multiple servers, it must provide a single system view to the user. Accordingly, each application must view the complete cluster as the single RM; do not consider individual instances as separate RMs.

When making a remote procedure call on a cluster, use the cluster name as the RM:

```
exec <cluster_name>.<dbname>.<owner>.<procedure_name>
```

Make sure that the transaction manager (TM) performing XA transactions on a cluster uses the cluster name as the logical resource manager (LRM) name in the XA configuration file. Do not use the instance name. For

more information about XA configuration, see the *SAP ASE XA Interface Integration Guide for CICS, Encina, and TUXEDO*.

Similarly, applications using jConnect to run XA transactions use the cluster name when creating XADataSource objects. For example:

```
XADataSource xads = (XADataSource)
    ctx.lookup("<server_name>=<cluster_name>");
```

Although the cluster is named as the RM, each transaction runs on a single instance (the owner instance) in the cluster, and the state of the transaction is maintained only on that instance. A running transaction cannot migrate between instances in the cluster. All Transact-SQL statements and transaction commit processing execute only on the instance that owns the transaction.

16.3 Handling Requests on Nonowner Instances

Some distributed transaction mechanisms, such as X/Open-XA protocols, are not connection-oriented, and allow applications to issue commands on different connections. As a result, requests to resume work on a suspended transaction branch can be issued on a nonowner instance.

Likewise, requests for commit processing can be issued on a connection to a nonowner instance. In SAP ASE Cluster Edition, such requests issued on nonowner instances require special processing.

When a request is issued on a nonowner instance, the SAP ASE Cluster Edition determines the owner instance for the transaction, and then migrates the connection to the owner instance with the help of the connectivity libraries. The connection migration is transparent to the application. After a successful connection migration, the request is processed on the owner instance.

ASTC transactions do not need connection migration. The ASTC mechanism requires that all Transact-SQL statements and the commit processing for transactions are issued on the same connection.

Related Information

Client Applications and Client/Server Interaction [page 71]

16.4 Handling Instance Failures

X/Open-XA and ASTC protocols do not allow the resource manager (RM) to unilaterally commit or abort a prepared transaction without instructions from the transaction manager (TM). The RM must retain a prepared transaction until it is apprised of the transaction's end status from the TM.

However, an instance in the cluster can fail when a transaction branch is running on it. The cluster coordinator reinstantiates the prepared transaction of the failed instance during failover recovery. Once the transaction is reinstantiated on the cluster coordinator, the TM can commit or roll back the transaction.

The commands that commit or roll back the reinstantiated transaction can land on an instance other than the current owner instance. In this scenario:

- For XA transactions – the server migrates the connection to the owner instance, which processes the request.
- For ASTC transactions – the instance receiving the `commit` or `rollback` command acts as a proxy by forwarding the request to the owner instance for processing and then sending the reply to the TM.

Only prepared transaction branches on the failed instance are reinstantiated during failover recovery. If a transaction branch running on a failed instance is not yet prepared, the SAP ASE Cluster Edition rolls back its work during failover recovery.

16.5 Transaction Coordination With ASTC

The SAP ASE Cluster Edition, like the nonclustered SAP ASE, uses the ASTC mechanism to coordinate a distributed transaction that spans multiple server installations. Transaction coordination services are transparent to the client executing the distributed transaction.

When a local client transaction executes an RPC or updates data via CIS, the coordination service creates a new transaction name for the remote work and propagates that transaction to the subordinate remote server. When the local client commits or rolls back the local transaction, SAP ASE Cluster Edition coordinates that request with each of the subordinate servers to ensure that the remote transactions are committed or rolled back as well.

The SAP ASE Cluster Edition executes special ASTC_HANDLER tasks to handle failures such as connection termination, a remote server failure, or a local server failure. The ASTC_HANDLER propagates the end status of the transaction to the remote participants after recovering from the failure condition. ASTC_HANDLER tasks are executed on each instance in the cluster. In the case of failures, an ASTC_HANDLER task on an instance propagates the end status to the remote participant only for the distributed transaction started on that instance. An ASTC_HANDLER task on the cluster coordinator has the additional work of handling the remote participants of the transaction from the previous start-up of the cluster or the remote participants on an instance when that instance leaves the cluster.

16.6 The Impact of Connection Migration

For XA transactions, the SAP ASE Cluster Edition uses connection migration to handle requests on nonowner instances.

16.6.1 Connection Migration and Performance

Connection migration can negatively affect performance, but you can reduce this affect.

Connection migration can negatively affect performance, because it requires:

- Terminating an existing connection on a nonowner instance.
- Creating a new connection to the instance running the transaction.
- Saving context information on the existing connection, and then restoring this information on the owner instance following connection migration.

To reduce the negative effect of connection migration on performance:

- Use logical clusters and workload manager capabilities to bind XA transactions to a specific instance in the cluster so that connection migration is unnecessary.
- Modify applications to avoid unnecessary disconnections and reconnections.

Related Information

Managing the Workload [page 98]

16.6.2 When Connections Cannot Migrate

On occasion, an application can issue a request on a connection that is in a state that does not allow migration.

A connection cannot migrate, for example, if an application connects to an instance, and before issuing the request to perform an operation on the transaction, uses the connection to create a temporary table. If such a connection is to a nonowner instance, SAP ASE cannot process the request, and the application receives a protocol-specific error code. In the case of X/Open-XA protocols, the TM receives XAER_RMERR as a return status, which indicates that the cluster has not processed the XA command. The transaction has not rolled back; it is still running on the owner instance.

To address this error, reissue the command on a different connection. XAER_RMERR is a generic error that is returned in many error situations. You can use the functions `xact_connmigrate_check` and `xact_owner_instance` to determine whether the XAER_RMERR error is due to a connection that cannot migrate.

To avoid XAER_RMERR errors caused by nonmigratable connections:

- Do not use the connections used for XA applications for general use, since a connection can be left in a state that does not allow it to migrate.
- Use trace flag 3960 to avoid an XAER_RMERR error for `commit` and `rollback` commands issued on nonmigratable connections to nonowner instances. When you use trace flag 3960, the nonowner instance acts as proxy between the TM and the owner instance by forwarding the request from the TM to the owner instance, and then sending the reply to the TM.

Related Information

Criteria for Migration [page 77]
Configuration and System Issues [page 140]

16.7 Configuration and System Issues

Use these tips to troubleshoot configuration and system issues.

- When configuring the cluster for DTM, specify the cluster name as the RM. Make sure your applications use the cluster name in ASTC RPCs and in the TM configuration.
- The DTM-relevant configuration parameters (`enable DTM`, `enable xact coordination`, and `strict dtm enforcement`) are cluster-static in the SAP ASE Cluster Edition. If you change the values of these parameters, restart the cluster for them to take effect.
- The `instanceid` column in the `syscoordinations` and `systransactions` system tables identifies that the owner instance of the transaction. An `instanceid` with a value of zero (0) indicates the owner instance has been shut down or has failed, and that the cluster coordinator is now the owner instance of the transaction.
- Two functions support DTM on clustered SAP ASE:
  - `xact_owner_instance(<XID>)` – returns the instance on which an external transaction is running.
  - `xact_connmigrate_check(<XID>)` – determines whether the current connection can process an external transaction.The `<XID>` parameter is optional.
    If you specify an external transaction name (`<XID>`):
    - Returns 1 if the connection is either to the owner instance, or to a nonowner instance and the connection can migrate.
    - Otherwise, returns 0.

    If you do not specify an `<XID>`:
    - Returns 1 if the connection can migrate.
    - Otherwise, returns 0.
- If a cluster instance is running a distributed transaction branch, do not use the polite `shutdown <instance_name>` command to shut down the instance. Instead, use:

  ```
  shutdown <instance_name> with nowait
  ```

  The polite `shutdown <instance_name>` command can roll back regular user transactions running on an instance, but it cannot roll back distributed transaction branches in the prepared state. This means you can use the polite instance `shutdown` only if there is no active distributed transaction on the instance. Shutting down the instance using the `nowait` option triggers a failover recovery that reinstantiates the prepared distributed transaction branches of the instance on the cluster coordinator.

  > **Note**
  >
  > Although you cannot use the polite `shutdown <instance_name>` command, you may use the polite `shutdown <cluster_name>` command, even if instances in the cluster have distributed transaction branches running on them.

For general information about configuring and administered DTM, see *SAP ASE Using Adaptive Server Distributed Transaction Management Features*.

Related Information

Handling Instance Failures [page 137]

17 Recovery

The SAP ASE Cluster Edition provides failover recovery, during which the database remains online and continues to be used. However, if you request data that was modified by the failed instance, and which must be recovered, the user process is blocked until the data is brought to a consistent state by the failover recovery process.

The SAP ASE Cluster Edition supports:

- Recovery that occurs at cluster start-up. This is the same as the nonclustered SAP ASE recovery that happens when you start the server during a cold restart.
- Failover recovery. This recovers data that a failed instance modified while other instances were using the same database in a shared-disk cluster.

Recovery begins when the cluster membership service detects a failure, then the recovery event handler on the coordinator instance recovers the system databases. Then the surviving instances (including the coordinator instance) recover the user databases in parallel. One of the PCM threads on each instance performs the recovery task. Other threads on the coordinator instance, as well as other instances, pursue other activities.

If trace flag 3483 is on, distributed failover recovery is disabled, and user databases are recovered in serial order by the recovery event handler on the coordinator instance.

The SAP ASE Cluster Edition blocks any process that attempts to access other processes that are trying to access data modified by the failed instance, until the recovery of the failed instance is completed.

The SAP ASE Cluster Edition recovers user databases in the order specified by their database IDs (`dbid`), and does not use any user-defined recovery order during instance failover recovery.

17.1 Recovery Process

The recovery process for the SAP ASE Cluster Edition is similar to that of a nonclustered SAP ASE.

Table 8: Recovery Steps for Clustered and Nonclustered Servers

Nonclustered SAP ASE Recovery Steps	SAP ASE Cluster Edition Recovery Steps
1. Estimate recoverable log boundaries – the recoverable log is from the oldest active transaction recorded in the recovery checkpoint to the end of the log	1. Estimate recoverable log boundaries (same as in nonclustered SAP ASE).
2. Analysis – scans the recoverable log forward, from the oldest active transaction to the end of the log, builds recovery information as incomplete transactions and so on. This information is used in the redo, undo, and post-undo passes.	2. Analyze the database's log being recovered (same as in nonclustered SAP ASE).

Nonclustered SAP ASE Recovery Steps	SAP ASE Cluster Edition Recovery Steps
3. Redo – scans the recoverable log forward, and re-executes operations specified by log records, as needed.	3. Redo. In addition, the locks for incomplete transactions are acquired.
	4. Reserve compensation log record (CLR) space to undo incomplete transactions on the failed instance.
	5. Release all locks acquired by the failed instance prior to the failure, except for those acquired at step 3 for incomplete transactions.
	6. Fill free space information (threshold manager recovery).
4. Undo – works from the end of the log back to the beginning of the oldest incomplete transaction, and undoes operations specified by log records for incomplete transactions. For each transaction, SAP ASE logs a CLR.	7. Undo. In addition, logical locks for incomplete transactions are released as completed.
5. Post-undo – logs a checkpoint record on the database, fills free space count recovery information (threshold recovery), and clears caches.	8. Post-undo, including flushing all dirty buffers on the recovery instance. The SAP ASE Cluster Edition does not perform a checkpoint after instance failover recovery.

Note

For information about the specific steps needed to complete the passes in this algorithm, see the SAP ASE *System Administration Guide: Volume 2*.

17.2 Single Transaction Log

There is only one log per database for the cluster, and it is logically partitioned. A log record marker for the SAP ASE Cluster Edition contains the instance ID, which it used internally to scan log records logged by specific instances, especially during failover recovery.

A checkpoint log record always has an instance ID with a value of zero, so it is always included in the log scan, regardless of the instance ID specified in the log scan.

17.2.1 Understanding Distributed Timestamps

When a page is updated, the next timestamp is taken from the database table and put on the page using `getnewts()`.

```
getnewts(){
```

```
dbts++; return dbts
}
```

Distributed recovery requires determining the highest timestamp in each instance, and finding the highest among those. As in a nonclustered SAP ASE recovery, all timestamps are initially set to the same value. That value is the highest timestamp across all the instances in a cluster.

18 Instance Recovery

SAP ASE Cluster Edition supports multiple simultaneous failures.

18.1 Single-Instance Recovery

The SAP ASE Cluster Edition provides failover recovery, during which time the database remains online and usable. However, if you request data that was modified by the failed instance, and which must be recovered, the user process is blocked until the data is brought to a consistent state by the failover recovery process.

The SAP ASE Cluster Edition supports:

- Recovery that occurs at cluster start-up, which is the same as the nonclustered SAP ASE recovery that happens when you perform a cold restart of the server.
- Failover recovery, which recovers data that a failed instance modified while other instances were using the same database in a shared-disk cluster. In this chapter, "failover" refers to an instance failover, rather than a client connection failover.

Recovery begins when the cluster membership service detects a failure. The recovery event handler on the coordinator instance recovers the system databases, then surviving instances (including the coordinator instance) recover the user databases in parallel. One of the PCM threads on each instance performs the recovery task. Other threads on the coordinator instance, as well as other instances, pursue other activities.

If trace flag 3483 is on, distributed failover recovery is disabled, and user databases are recovered in serial order by the recovery event handler on the coordinator instance.

Processes trying to access data modified by the failed instance are blocked until recovery is complete.

The SAP ASE Cluster Edition recovers user databases in the order specified by their database IDs (`dbid`), and does not use any user-defined recovery order during instance failover recovery.

18.1.1 Single Transaction Log

There is only one log per database for the cluster, and it is logically partitioned. A log record marker for the SAP ASE Cluster Edition contains the instance ID, which internally scans log records logged by specific instances, especially during failover recovery.

A checkpoint log record always has an instance ID with a value of zero, so that it is always included in the log scan, regardless of the instance ID specified in the log scan.

18.2 Multiple Simultaneous Failures

The SAP ASE Cluster Edition supports multiple simultaneous instance failures, when more than one instance fails within a single cluster view, but the cluster remains online and provides the same failover recovery as it does when a single instance fails.

The SAP ASE Cluster Edition provides a configuration parameter, `cluster redundancy level`, that allows a database administrator to set the maximum number of recoverable simultaneous instance failures for the cluster.

18.2.1 Enabling Multiple Simultaneous Failovers

The number of recoverable failed instances is the cluster redundancy level (CRL).

The CRL is the maximum number of instances that can fail simultaneously while allowing recovery to proceed concurrently with other activities. If the number of failed instances exceeds the maximum number specified by the configuration parameter, the cluster shuts down.

When you configure the CRL with a number greater than 1, the lock replication in the cluster is also increased. The CRL is at the physical cluster level, and always takes precedence over a failover logical cluster configuration; so your CRL cannot exceed the number of physical instances in your cluster.

The `cluster redundancy level` configuration parameter is a static, cluster parameter with a default value of 1. Its minimum allowed value is 1, and its maximum allowed value is one less than the `'maximum number of instances'` values that you specified in the `cluster.cfg` or quorum file. The value of `cluster redundancy level` remains the same, even if one or more instances leave the cluster.

The syntax is as follows, where `<config_value>` is the number of instances that can fail simultaneously::

```
sp_configure 'cluster redundancy level', <config_value>
```

Failover recovery recovers data modified by a failed instance while other instances are using the same database. With a CRL of 1 (the default), SAP ASE maintains at least two copies of each lock in the cluster. Thus, if one instance fails, one copy of the lock remains, so that SAP ASE can mark the data modified by the failed instance. Similarly, with a CRL of "n", SAP ASE has at least "n+1" copies of each lock in the cluster, and can perform failover recovery for "n" simultaneous failovers.

The cluster cannot start if you set either of these:

- The value of `maximum number of instances` to a value that is equal to or less than the value of `cluster redundancy level`
- The value of `cluster redundancy level` to a value equal to or greater than the value of `maximum number of instances`

Set `cluster redundancy level` before starting the cluster. Restart the cluster if you change its value during runtime.

Set a `cluster redundancy level` of 1 unless multiple instance failure is common.

18.2.1.1 Performance Limitations and the Cluster Redundancy Level

When the CRL value increases, the number of simultaneous instance failures a cluster can support also increases. However, an increase in the CRL value also causes an increase in messaging traffic, because there are multiple copies of a lock in the cluster.

An increase in overhead is required to maintain this redundancy level.

For example, other configuration parameters such as `number of locks` and `cache size` also need more resources for CRL values greater than 1; increase `max memory` for the same `number of locks` value. CRL is resource-intensive, even though SAP ASE performs the overhead value computations transparently.

18.3 Recovery Algorithm

The internal database recovery process in the SAP ASE Cluster Edition is similar to that of a nonclustered SAP ASE. The same phases are necessary for single and multiple instance failure recovery.

Table 9: Recovery Steps for Clustered and Nonclustered Servers

Nonclustered SAP ASE Recovery Phases	SAP ASE Cluster Edition Recovery Phases
1. Estimate recoverable log boundaries – the recoverable log is from the oldest active transaction recorded in the recovery checkpoint to the end of the log.	1. Estimate recoverable log boundaries (same as in nonclustered).
2. Analysis – scans the recoverable log forward, from the oldest active transaction to the end of the log, builds recovery information as incomplete transactions and so on. This information is used in the redo, undo, and post-undo passes.	2. Analyze the log from the database being recovered (same as in nonclustered SAP ASE).
3. Redo – scans the recoverable log forward, and re-executes operations specified by log records, as needed.	3. Redo – the locks for incomplete transactions are acquired.
	4. Reserve compensation log record space to undo incomplete transactions on the failed instance.
	5. Release all locks acquired by the failed instance prior to the failure, except for those acquired at step 3 for incomplete transactions.
4. Undo – works from the end of the log back to the beginning of the oldest incomplete transaction, and undoes operations specified by log records for incomplete transactions. For each transaction, SAP ASE logs a compensation log record.	6. Undo – also, logical locks for incomplete transactions are released as completed.

Nonclustered SAP ASE Recovery Phases	SAP ASE Cluster Edition Recovery Phases
	7. Fill free space information (threshold manager recovery).
5. Post-undo – logs a checkpoint record on the database, fills free space count recovery information (threshold recovery), and clears caches.	8. Post-undo – flushes all dirty buffers on the recovery instance. The SAP ASE Cluster Edition does not perform a checkpoint after instance failover recovery.

Note

For more information, see the *SAP ASE System Administration Guide: Volume 2*.

19 Single Instance Databases

In SAP ASE Cluster Edition, you can create databases that allow access from a single node, which improves performance by removing cluster locks. These restricted databases are called single instance databases.

Restricting access to a single node reduces the impact of cluster infrastructure overheads for locking, messaging, data synchronization and so on, improving performance compared to traditional cluster databases. You can modify data on any node in the cluster on regular databases. SAP ASE uses cluster-wide locks to serialize any access or modification of data pertaining to normal databases. Because single-instance databases restrict access to a single node, they do not require clusters locks, resulting in improved performance over regular cluster databases.

Along with performance improvements, single-instance databases provide the same high availability benefits associated with traditional cluster databases: they are automatically brought online after a failure, and incoming connections are redirected automatically to the new owner node.

You can relocate single-instance databases easily within the cluster for load balancing or maintenance. If the single-instance database outgrows the resources available on its owner node, it can be relocated to a more powerful machine within the cluster or converted to a traditional cluster database.

Each single-instance database must be associated with a single-instance logical cluster, which is an abstract representation of a single instance in a physical shared-disk cluster. The single-instance logical cluster manages failover and client application access to the cluster. Single-instance logical clusters need not manage workload because they are online only on single nodes. A single-instance logical cluster is configured for a single primary node, but can have either multiple, or no, failover nodes.

All single-instance logical clusters have an owner or primary instance, which is the instance to which the single-instance logical cluster is assigned. The single-instance database associated with this single-instance logical cluster is accessible only from this instance. If a single-instance logical cluster is online on an instance, this instance is its owner. If the single-instance logical cluster is not online, its primary node is its owner.

A single-instance logical cluster can have at most one primary node, and can be online only on one node at a time.

You cannot directly configure the ownership and failover behavior of a single-instance database. Instead, bind the single-instance database to a single-instance logical cluster, and use the single-instance logical cluster infrastructure to configure ownership and failover behavior.

19.1 Binding Single-Instance Databases to Single-Instance Logical Clusters

Single-instance databases are bound to a single-instance logical cluster to automatically fail over and redirect clients to the new database owner after the original database owner fails.

When you configure single-instance logical clusters, assign the primary node as the owner of the associated single-instance database. The single-instance logical cluster brings the single-instance database online on this primary node. During failover, the single-instance logical cluster determines the next available node from the

list of failover nodes, which it configures as the owner of the associated single-instance database and brings the single-instance database online on it.

Clients connect to a single-instance database through the single-instance logical cluster to which it is bound. Clients are guaranteed to connect to the current database owner when they connect to the single-instance logical cluster associated with the single-instance database, even if the node fails.

The database administrator can determine to which failover instance the single-instance logical cluster fails over, and can force the single-instance logical cluster to manually failback to the previous primary node once it is back online using the single-instance logical cluster interface using `sp_cluster`. See the *SAP ASE Reference Manual: Procedures*.

All single-instance logical clusters have an owner or primary instance, which is the instance to which the single-instance logical cluster is assigned, and the single-instance database associated with this single-instance logical cluster is accessible only from this instance. If a single-instance logical cluster is online on an instance, this instance is its owner. If the single-instance logical cluster is not online, its primary instance is its owner.

A single-instance logical cluster can have at most one primary node, and can be online only on one node at a time.

19.2 Configure a Cluster for a Single-Instance Database

To configure the cluster for a single-instance database, create the database and the logical cluster, set the affinity for the logical cluster, and bring the database online.

Procedure

1. Create the single-instance database:

   ```
   create database <database_name> for single instance access
   ```

   This example creates a single-instance database named SIDB1:

   ```
   create database SIDB1 for single instance access
   ```

   > **Note**
   >
   > You cannot access this database until you bind it to an online single-instance logical cluster.

2. Create the single-instance logical cluster:

   ```
   sp_cluster logical, "create", <lc_name>, "for single instance access"
   ```

   By default, creating the logical cluster sets these parameters:
   - `down_routing` – `disconnect`
   - `failover` – `instance`

- `fail_to_any - off`

See the *SAP ASE Cluster Edition Users Guide* and the *SAP ASE Reference Manual: Procedures*.

This example creates a single-instance logical cluster named LC1:

```
sp_cluster logical, "create", LC1, "for single instance access"
```

3. Add the primary instance to the single-instance logical cluster:

```
sp_cluster logical, "add", <lc_name>, "instance", <instance_list>
```

This example adds the N1 instance to the LC1 single-instance logical cluster:

```
sp_cluster logical, "add", LC1, "instance", N1
```

4. Add the secondary instance to the single-instance logical cluster:

```
sp_cluster logical, "add", <lc_name>, "failover", <instance_name>
```

This example adds the N2 instance to the LC1 single-instance logical cluster:

```
sp_cluster logical, "add", LC1, "failover", N2
```

5. Bring the single-instance logical cluster online:

```
sp_cluster logical, "online", <lc_name>
```

This example brings the single-instance logical cluster online:

```
sp_cluster logical, "online", LC1
```

If a single instance database is already bound to this single instance logical cluster, it is also brought online.

6. Bind the single-instance database to the single-instance logical cluster:

```
sp_cluster logical, "add", <lc_name>, "database", <database_name>
```

If the single-instance logical cluster is online, this system procedure brings the single-instance database online on the primary node for the single-instance logical cluster, and configures the primary node as the owner of the single-instance database. If the single-instance logical cluster is not online, the single-instance database is brought online when you configure the primary node for the single-instance logical cluster and bring it online.

If the single-instance database is already bound to another single-instance logical cluster, `sp_cluster` removes the earlier binding, shuts down the single-instance database, and configures the new binding. `sp_cluster` brings the single-instance database online when the corresponding single-instance logical cluster comes online.

19.3 Administering Single-Instance Databases

Use the `sp_cluster` system procedure and the `alter database` command to administer single-instance databases and logical clusters.

19.3.1 Remove a Binding

Use `sp_cluster` to remove a binding.

Procedure

Use this syntax to remove the binding between a single-instance database and a single-instance logical cluster:

```
sp_cluster logical, "drop", <lcname>, "database", <database_name>
```

`sp_cluster` does not shut down the single-instance database; it remains on the same node, and is available for connections. If this node subsequently fails, the single-instance database is recovered, but is not assigned an owner.

19.3.2 Migrate a Single-User Database to Another Node

Use `sp_cluster` to migrate a single-instance logical cluster and its single-instance logical cluster to another node (often necessary for maintenance work).

Procedure

Use this syntax to temporarily failover a single-instance logical cluster to another node:

```
 sp_cluster logical, "failover", <lcname>, "instance", <from_instance_list>,
<to_instance_list>[, <wait_option>[,<timeout>[, "sync"[, @handle output]]]]
```

If the cluster restarts while the single-instance logical cluster is running on the new node, the cluster brings the single-instance logical cluster online on its primary node. See the *SAP ASE Cluster Edition Users Guide* and the *SAP ASE Reference Manual: Procedures*.

19.3.3 Migrate the Primary Instance

Permanently change the owner of the single-instance database to the primary instance to which the single-instance logical cluster is bound.

Procedure

1. Take the single-instance database offline:

```
sp_cluster logical, "offline", <lc_name>,
   {cluster | instance, <instance_list> }
   [, <wait_option>[, <timeout>,[, @handle output ]]]
```

2. Drop the current primary instance from the logical cluster:

```
sp_cluster logical, "drop",  <lc_name>,
   {cluster | instance, <instance_name> |
    failover, <instance_list> | route, <route_type>, <key_list>}
```

3. Add a new instance as the primary instance:

```
sp_cluster logical, "add", <lc_name>,
   {route, <route_type>, <key_list> | instance, <instance_name> | failover,
<instance_list>}
```

4. Bring a new instance online:

```
sp_cluster logical, "online", { <lc_name>[, <instance_list>]}
```

See the *SAP ASE Cluster Edition Users Guide* and the *SAP ASE Reference Manual: Procedures*.

19.3.4 Remove Single-Instance Databases

Use `drop database` to remove single-instance databases, and `sp_cluster` to remove single-instance logical clusters.

Procedure

1. Use `drop database` to remove single-instance databases. You must drop a single-instance database from its primary node. If a single-instance database has no primary node, drop it from any node.

   This example removes the `SIDB1` single-instance database:

```
drop database SIDB1
```

   SAP ASE automatically removes the logical cluster bindings.

2. Remove the single-instance logical cluster. This example removes the `LC1` single-instance logical cluster:

```
sp_cluster logical, "drop", LC1, "cluster"
```

19.3.5 Convert a Regular Cluster Database to Single-Instance Database

Use `alter database` to convert a cluster database, and `sp_cluster` to add a single-instance database to a single-instance logical cluster.

Procedure

1. Use `alter database` to convert the cluster database. This example converts the `SIDB1` database:

```
alter database SIDB1 for single instance access
```

   See the *SAP ASE Reference Manual: Commands*.
2. Use the `sp_cluster` system procedure to add the single-instance database to a single-instance logical cluster. This example adds the `SIDB1` database to the `LC1` logical cluster:

```
sp_cluster logical, "add", LC1, "database", SIDB1
```

   See the *SAP ASE Reference Manual: Procedures*.

19.3.6 Converting a Single-Instance Database to a Regular Cluster Database

Use `alter database` to convert a single-instance database to a regular cluster database.

This example converts the `SIDB1` database:

```
alter database SIDB1 for clusterwide access
```

See the *SAP ASE Reference Manual: Commands*.

19.4 Limitations

Single-instance databases include limitations.

The owning node is the only node that allows you to issue these commands:

- `load database`
- `dump database`
- `load transaction`
- `dump transaction`
- `mount database`
- `unmount database`

- Single-instance databases support replication. However, you must start the Replication Agent for a single-instance database from the owning node. During manual or automatic failover, the Replication Agent is automatically halted on the original owner and restarted on the new primary node during the failover and recovery process.
- Single-instance databases are unavailable while they are relocating to another node.
- Single-instance database may recover from failover more slowly than comparable traditional cluster databases.
- Single-instance databases do not automatically come online if the single-instance logical cluster was not configured for `auto-online`, and the base node of the single-instance logical cluster does not come online when you restart the cluster.

20 Temporary Databases

You can create and manage temporary databases, as well as bind logins to a temporary database or group of temporary databases. You can also create and manage private devices, and use them for local user temporary databases.

Temporary databases provide storage for temporary tables and other temporary working storage needs. The SAP ASE Cluster Edition supports both local and global temporary databases. Local temporary databases can be accessed only by the owning instance, and are used primarily to store session-specific private temporary objects such as #tables, worktables, and fake tables. Global temporary databases can be accessed by all instances in the cluster and are used to store temporary objects that can exist beyond the current session.

The SAP ASE Cluster Edition supports:

- Local system temporary databases
- Local user temporary databases
- Global user temporary databases
- The global system temporary database (`dbid` of 2)

Temporary databases are inherited from the `model` database with the following database options set:

- `select into/bulkcopy`
- `trunc log on chkpt`

A guest user is automatically added to the temporary database, and all users are granted `create table` permission.

20.1 Local Temporary Databases

The SAP ASE Cluster Edition lets you create temporary databases for each instance in the cluster. An instance-specific temporary database is called a local temporary database; the instance owning the local temporary database is called its owner instance.

Each local temporary database is bound to a particular instance and can be accessed only from that instance. You create a local system temporary database for each instance in the cluster. Creation of local user temporary databases is optional.

20.1.1 Local User Temporary Databases

You create multiple local user temporary databases for each instance and then bind applications or logins to an individual local user temporary database or group of temporary databases.

You create a local user temporary database on a shared or nonshared device. After creation, each local temporary database is added to the owning instance's list of available databases. Other instances in the

cluster may or may not be able to physically access the device, but only the owning instance actually accesses it.

20.1.2 Local System Temporary Databases

A local system temporary database is the required default temporary database for each instance.

You configure it at the time that you configure the cluster, or when a new instance is added to the cluster. The instance stores all session-specific temporary objects (such as # tables and worktables) in this database unless you create and use local user temporary databases for the instance. You create local system temporary databases on shared storage.

In a nonclustered SAP ASE environment, the system temporary database `tempdb` (`dbid` 2) is added to the default temporary database group. In the SAP ASE Cluster Edition, the local system temporary database is not part of the default group for the instance. The local system temporary database is assigned to a session only if the default group for the current instance is empty and no other bindings have been specified.

> **Note**
>
> For the SAP ASE Cluster Edition, the default temporary database assigned to a session is a local temporary database for the instance, not the system `tempdb` (with a `dbid` of 2). You may need to modify applications that, in a nonclustered environment assumed the default assigned temporary database as system `tempdb` (with a dbid of 2) in their actions, so that these actions are now applied to the assigned local temporary database.
>
> For example, in a nonclustered SAP ASE, if an application truncates the log of the default temporary database with the`dump tran tempdb with truncate_only` command, then in the SAP ASE Cluster Edition, you modify the application to truncate the log of the default assigned temporary database:
>
> ```
> declare @tempdbname varchar(30)
> select @tempdbname = db_name(@@tempdbid)
> dump tran @tempdbname with truncate_only
> ```

Related Information

Using Private Devices for Temporary Data [page 166]

20.1.2.1 Separate User Log Cache for a Session's Temporary Database

The SAP ASE Cluster Edition supports multiple temporary databases, such as user local temporary databases, system local temporary databases, and global temporary databases (`tempdb`).

The SAP ASE Cluster Edition supports a separate user log cache only for the session's temporary database, and not for any other temporary databases.

The session's temporary database can be:

- A user local temporary database
- A system local temporary database
- A global temporary database when the cluster is in `tempdb` configuration mode

During `tempdb` configuration, when local system or local user temporary databases do not exist, the session's default temporary database is the system `tempdb` (with a database ID of 2). You cannot use a separate user log cache for the system `tempdb` during its configuration.

20.2 Global Temporary Databases

The system automatically creates a global system temporary database (`dbid` 2) when the master device is created. You can create additional global temporary databases.

You can access global temporary databases from any instance in the cluster, which allows you to create temporary objects that can be shared across sessions on the same or different instances. You cannot create temporary objects such as # tables and worktables in them, because global temporary databases cannot be assigned to a specific session.

Global temporary databases are identical to normal user databases with one exception: they are re-created every time the cluster starts. Global temporary databases provide standard logging, log flushing, I/O behavior, and runtime rollback features. They are supported primarily for backward compatibility with earlier versions of SAP ASE.

All objects in the global temporary database are lost when the cluster shuts down. When an instance fails, the instance failover recovery step recovers the objects in global temporary databases as well, which ensures that objects created during a session on the failed instance continue to be available from surviving instances. The global temporary database ensures that temporary shareable workspace continues to exist as long as even one instance in the cluster is active.

> **Note**
>
> Applications created on SAP ASE may create shareable tables in user-created temporary databases. To enable existing applications to continue to work with the SAP ASE Cluster Edition, drop these user-created temporary databases that include these shareable tables and re-create them as global temporary databases with the same names.

20.3 Characteristics of Temporary Databases

Temporary databases include local system, local user, global user, and global system temporary databases..

Table 10: Characteristics of Temporary Databases

Feature	Local System Temporary Database	Local User Temporary Database	Global User Temporary Database	Global System Temporary Database
Supported temporary database objects	Both session-specific temporary objects and regular database objects	Both session-specific temporary objects and regular database objects	Regular database objects only, no session-specific temporary objects	Regular database objects only, no session-specific temporary objects*
Recovery	Re-created when owning instance restarts	Re-created when owning instance restarts	Re-created when the cluster restarts, and transactionally recovered when an instance fails	Re-created when the cluster restarts, and transactionally recovered when an instance fails
Accessibility	Accessible from owning instance only, but it must be created on a shared device so that it can be created and dropped from other instances	Accessible from owning instance only	Accessible from any instance	Accessible from any instance
Creation	User-created, one (required) for each instance	User-created, zero or more for each instance	User-created, zero or more for the cluster	System-generated (dbid = 2), one for the cluster
Binding allowed?	No	Yes	No	No
Storage (shared or private devices)	Shared storage only	Both shared and local storage	Shared storage only	Shared storage only

* During `tempdb` configuration mode, the global system temporary database acts like a local system temporary database to the boot coordinator, supporting both session-specific temporary objects and regular database objects.

20.4 Creating Temporary Databases

In the SAP ASE Cluster Edition, each cluster has a global system temporary database, and each instance in the cluster has a local system temporary database.

Other temporary databases are optional. The global system temporary database is created when you install SAP ASE. This section describes how to create the other temporary databases.

> **Note**
>
> Create temporary databases that are of a similar size to allow applications to access each instance without regard to temporary space requirements.

20.4.1 Creating Local System Temporary Databases

Create local system temporary databases using the SAP Control Center or `sybcluster`.

See the SAP ASE Cluster Edition installation guide for your platform.

Create a local system temporary database for each instance during the initial start-up of the cluster and later on whenever you add an instance to the cluster. You can create the local system temporary database on a shared device. You can create or drop a local system temporary database from any instance, but you can access it only from the owning instance.

20.4.2 Creating Local and Global User Temporary Databases

You can create local and global user temporary databases at any time.

- Create a local user temporary database from the owning instance. For example, to create the local temporary database "local_temp1" on "ase1", use:

```
create temporary database local_temp1
    for instance ase1 on default = 50
```

- Create a global user temporary database from any instance in the cluster. For example, to create the global temporary database "global_tempdb1" on "ase1", log in to "ase1", use::

```
create global temporary database global_tempdb1
    on default = 50
```

> **Note**
>
> If you do not include the optional `for instance <instance_name>` parameter in the `create database` statement, SAP ASE creates the temporary database for the current instance. See the *SAP ASE Reference Manual: Commands* for a description of its complete syntax.

20.5 Binding Users and Applications to Temporary Databases

Bind logins and applications to temporary databases or temporary database groups, and create temporary database groups in both the nonclustered SAP ASE and the SAP ASE Cluster Edition using `sp_tempdb`.

In the SAP ASE Cluster Edition, which supports both local and global temporary databases, you can bind logins or applications to local user temporary databases or temporary database groups, but not to global temporary databases.

The `default` temporary database group is system-generated and always present, but is empty unless you use `sp_tempdb` to add local temporary databases to it.

> **Note**
>
> In the SAP ASE Cluster Edition, the system temporary database `tempdb` (`dbid` 2) is a global system temporary database, and is not a member of the `default` group. This behavior differs from that of the system temporary database `tempdb` (`dbid` 2) in the nonclustered SAP ASE, in which it is, by default, a member of the `default` group.

The `sp_tempdb` user interface includes the option `unbindall_gr`, which removes all bindings to the specified group in either the nonclustered or clustered SAP ASE. In addition, the `unbind` option includes an `<instance_name>` parameter that is specific to the SAP ASE Cluster Edition.

20.5.1 Creating and Managing Temporary Database Groups

Creating and managing temporary database groups is the same for all editions of SAP ASE.

> **Note**
>
> In the SAP ASE Cluster Edition, the local system temporary database is not a member of the `default` group.

- Create a temporary database group using `sp_tempdb create`. This example creates a temporary database group named "tempdbgroup1":

  ```
  sp_tempdb "create", "tempdbgroup1"
  ```

- Drop a temporary database group using `sp_tempdb drop`. This example drops the temporary database group named "tempdbgroup1":

  ```
  sp_tempdb "drop", "tempdbgroup1"
  ```

- Add a database to a temporary database group using `sp_tempdb add`. This example adds database "local_tempdb1" to the temporary database group named "tempdbgroup1":

  ```
  sp_tempdb "add", "local_tempdb1", "tempdbgroup1"
  ```

- Remove a database from a temporary database group using `sp_tempdb remove`. This example removes database "local_tempdb1" from the temporary database group named "tempdbgroup1":

```
sp_tempdb "remove", "local_tempdb1", "tempdbgroup1"
```

See the *SAP ASE Reference Manual, Procedures* for complete syntax and usage information for `sp_tempdb`.

20.5.2 What You Can Bind

There are restrictions for binding a user login or application to databases or database groups.

- A user login or application can bind to multiple local user temporary databases, but can only bind to one database per instance.
  For example, you can bind “sa” to "local_tempdb1" on “ase1” and "local_tempdb2" on “ase2”, but you cannot also bind “sa” to "local_tempdb3" on “ase1”.
- A user login or application to only bind to one database group. Temporary database groups are visible from all instances.
  For example, if you bind `isql` to the `default` group, you cannot also bind `isql` to another temporary database group.
- A user login or application can bind to individual temporary databases or a temporary database group, but not to both.
  For example, if “sa” is bound to "local_tempdb1" on “ase1”, it cannot also be bound to the `default` group.

20.5.3 How the Session Binding is Determined

When a session is initiated, SAP ASE Cluster Edition checks to see whether a binding is applicable. If more than one binding is applicable, the SAP ASE Cluster Edition determines which binding is used based on the following algorithm. Only one binding is supported per session.

1. If a binding exists for the current login, that binding is used. Otherwise, the SAP ASE Cluster Edition searches for a binding for the current application. If both are found, SAP ASE uses the binding for the login, since a login binding takes precedence over an application binding.
   For example, if user “sa” is bound to "tempdb1" on instance “ase1”, and `isql` is bound to "tempdb2" on “ase1”, then, when user “sa” uses `isql` to initiate a session on “ase1”, only the “sa” user binding is used.

   > **Note**
   >
   > If the first binding fails, the SAP ASE Cluster Edition automatically assigns a member of the `default` group. The SAP ASE Cluster Edition does not search for another binding.

2. If the binding is to a database, the database is assigned. If the binding is to a database group, then SAP ASE assigns a member database from the current instance, selected in a round-robin fashion. Local temporary databases that are members of a group can be assigned only to a session established on the owner instance, not to a session on another instance using the group binding.
3. If a binding is not found or an assignment cannot be made, an assignment is attempted for a member of the `default` group, selected in a round-robin fashion.

4. If an assignment cannot be made to a member of the `default` group, the assignment is made to the local system temporary database.

20.5.4 Creating and Managing Bindings

Bind a login or application to a local temporary database or database group using `sp_tempdb bind`.

- To bind the "sa" login to the `default` group, enter:

```
sp_tempdb "bind", "LG", "sa", "GR", "default"
```

- To bind `isql` to "local_tempdb1" on "ase1", enter:

```
sp_tempdb "bind", "AP", "isql", "DB", "local_tempdb1"
```

- To remove one or more bindings, use `sp_tempdb unbind`. For the SAP ASE Cluster Edition, this command includes an `instance_name` parameter that lets you drop the binding to a particular local temporary database. This example unbinds `isql` from "local_tempdb2" on "ase2":

```
sp_tempdb "unbind", "AP", "isql", NULL, "ase2"
```

> **Note**
>
> If `isql` is bound to a database group, the above command removes the binding for the group. If multiple database bindings exist for `isql`, the above command removes only the binding for "ase2"; bindings to temporary databases on other instances are unaffected.

- To unbind a user login or application from a database group or any temporary databases, use the `unbind` parameter with only the login or application name. This example removes the binding for the "sa" login:

```
sp_tempdb "unbind", "lg", "sa"
```

- To unbind all logins and applications to a particular group, use `sp_tempdb unbindall_gr`. This example removes all bindings to "tempdbgroup1":

```
sp_tempdb "unbindall_gr", "tempdbgroup1"
```

- To unbind all login and application bindings to a particular database, use `sp_tempdb unbindall_db`. This example unbinds all bindings to "localtempdb1":

```
sp_tempdb "unbindall_db", "localtempdb1"
```

20.5.5 Displaying Group and Binding Information

Use `sp_tempdb show` to display a list of existing groups, group members, and login and application bindings.

This example displays the members of the temporary database group "tempdbgroup1" and their owner instances:

```
sp_tempdb "show", "gr" "tempdbgroup1"
```

To display a list of active sessions assigned a temporary database, use `sp_tempdb who`. This example displays active session assigned the temporary database "localtempdb1":

```
sp_tempdb "who", "localtempdb1"
```

20.5.6 Dropping Temporary Databases

Restrictions apply to dropping temporary databases.

20.5.6.1 Dropping All but the Last Local System Temporary Database

A local system temporary database is always in use if its owner instance is running. This means that in order to drop a local system temporary database, you need to shut down the instance, then drop the local system temporary database from another instance.

20.5.6.2 Dropping the Last Local System Temporary Database

Drop the local system temporary database of the last instance.

Procedure

1. Mark the database to be dropped using `sp_tempdb_markdrop`. In this example, "ase3_tdb1" is the last local system temporary database:

   ```
   sp_tempdb_markdrop ase3_tdb1
   ```

2. Shut down and restart the last instance. The local system temporary database marked for "drop" is not used.
3. Drop the temporary database from this instance.

20.6 Restrictions for Temporary Databases

Restrictions apply to temporary databases.

- Attempting to access a local temporary database owned by another instance results in an error message similar to:

  ```
  Error Number : 969
  You can access database '<local-tempdbname>' only from its owner instance
        '<owner-instance-name>'. You cannot access local temporary databases
  from
        non-owner instances except to use CREATE DATABASE and DROP DATABASE
  with local system
        temporary databases.
  ```

  In general, execute stored procedures and commands that access a local temporary database from the owner instance. Alternatively, you can use these methods to execute the operation on a local temporary database of a remote instance:

  - Use `connect to <instance_name>` to connect to the owner instance before you access any of the local temporary databases.
  - Use `sp_remotesql` to send the Transact-SQL statement to the remote instance. This example executes a query on local temporary database "local_tempdb_ase1" owned by instance `ase1`:

    ```
    sp_remotesql "ase1", "select * from local_tempdb_ase1..sysobjects"
    ```

  - For stored procedures, you can execute the procedure specifying the instance name. This example executes `sp_spaceused` on "local_tempdb_ase1" for instance “ase1”:

    ```
    ase1.local_tempdb_ase1.dbo.sp_spaceused
    ```

    Note

    Use `db_instanceid` to determine the owner instance ID of a local temporary database.

- Some stored procedures and commands that perform operations on all databases in a nonclustered SAP ASE, may skip these operations on remote local temporary databases in the SAP ASE Cluster Edition. To make sure these operations occur in all databases, execute the operation again for each of the skipped databases from the owner instance.
  For example, if you execute `sp_dbcc_faultreport` without specifying a database name, it skips remote local temporary databases from the fault report.
- You cannot grant or revoke user-defined roles to users of a local temporary database.
- You cannot grant or revoke permissions on objects in a local temporary database.
- You cannot use a local temporary database as the default database for any login; any attempt to do so causes `sp_addlogin` and `sp_modifylogin` to fail.
- You cannot include a referential integrity constraint that references a column on a local temporary database except from a table on the same local temporary database. `create table`, `alter table`, and `create schema` fail when they attempt to create a reference to a column on a local temporary database from a different database.
- You cannot encrypt a column with an encryption key located in a local temporary database unless the table resides on the same local temporary database. `create table` and `alter table` fail when they attempt to encrypt a column with an encryption key located on the local temporary database and the table resides in another database.

- When you drop a role, SAP ASE checks whether there are any references to this role in any database.
- When you create a role, SAP ASE checks if there are any users with the same name in any database.
- You cannot use encryption keys stored in a local temporary database to encrypt columns of other databases.
- You cannot specify database recovery order for any local system temporary database. Local system temporary databases are user-created, but they are always recovered with system databases.
- You cannot use `sp_configure` to change configuration options from inside the local temporary database context.

20.7 Private Device Support for Local Databases

The SAP ASE Cluster Edition allows you to use inexpensive storage, such as local disks and file system devices, to provide instance-specific temporary space needs.

- You can use such storage, created as a private device, to create instance-specific local user temporary databases. However, do not use private devices for local system temporary databases. Local system temporary databases must be created on shared storage.
- In the SAP ASE Cluster Edition, a private device is owned and accessed by one cluster instance only. That a device is private is a *logical* attribute; the physical path to the device may or may not be accessible from a non-owner instance.
- In the `sysdevices` table, a private device is identified by the bitmap `status2` column, and the instance that owns the device by the `instanceid` column of the row belonging to the device.
- An instance can own many private devices, but a private device is owned by a single instance.
- A private device can be used only for local user temporary databases, on the instance that owns the device. It cannot be used as default storage; if you invoke `sp_diskdefault` on a private device, the stored procedure fails and an error message appears.
- A private device cannot be mirrored, unmirrored, or remirrored.

20.7.1 Using Private Devices for Temporary Data

To achieve optimal performance of temporary data in the SAP ASE Cluster Edition, use fast local disks for local user temporary databases.

Create local system temporary databases on shared storage. To limit the use of shared storage for temporary data:

1. Create one or more private devices on local disks.
2. Create one or more local user temporary databases on the private devices.
3. Add these local user temporary databases to the `default` temporary database group.
   A user connection that has no explicit temporary-database binding is assigned a member database from the `default` group. This limits the use of local system temporary databases created on shared storage to internal system tasks while all user connections use local user temporary databases created on fast local disks.

20.7.2 Creating Private Devices Using disk init

Use the `disk init` command to create a private device.

The `disk init` command includes an optional `instance` parameter that marks the device as private to an instance.

This example creates a device named "private_dev1" that is private to the "cluster1_1" instance:

```
disk init
name = "private_dev1",
physname = "/usr/u/sybase/data/private_dev1.dat",
vdevno = 2, size = 5102, instance = "cluster1_1"
```

You can execute `disk init` from any of the instances in the cluster. The command is shipped to the owning instance internally. The owning instance, however, must be up and running to process the command, or else the command fails and the private device is not created.

20.7.3 Reinitializing Private Devices Using disk reinit

As part of the procedure for restoring the master database, restore the private device entries in the master database `sysdevices` table using `disk reinit`.

The `disk reinit` command includes the optional `instance` parameter, which marks a device as private to an instance.

This example restores the "private_dev1" device as a private device for the instance "cluster1_1" in the master database `sysdevices` table:

```
disk reinit
name = "private_dev1"
physname = "/usr/u/sybase/data/private_dev1.dat"
vdevno = 2, size = 5120, instance = "cluster1_1"
```

Execute this command from any instance in the cluster. If the owning instance is up and running, the command is shipped to it.

If the owning instance is not up and running, `disk reinit` inserts a row in `sysdevices`, corresponding to the device, but does not activate it until the instance is running. This is useful when devices must be restored before issuing `disk refit`.

> **Note**
>
> Take extra care regarding the parameters you pass when you restore a private device for an instance that is not running. SAP ASE cannot verify the physical path of a private device in such cases; it is assumed that the user provides a valid physical path.

20.7.4 Dropping Private Devices Using sp_dropdevice

Use `sp_dropdevice` to drop a private device.

You can execute this procedure from any instance in the cluster, but it fails to drop a private device if the owning instance is not running. For details about using `sp_dropdevice`, see the *SAP ASE Reference Manual: Procedures*.

20.7.5 Displaying Private Device Information Using sp_helpdevice

Use the `sp_helpdevice` system procedure to find out whether a device is private.

If the device is private, `sp_helpdevice` prints the name of the instance in the cluster that owns the private device.

In this example, "regular_dev" is a regular shareable device, while "private_dev1" is a private device owned by instance "cluster1_1":

```
1> sp_helpdevice
2> go
 device_name
    physical_name
    description
    status cntrltype vdevno      vpn_low     vpn_high
 ----------
    ----------------------------------------------------------------------------
    ----------------------------------------------------------------------------
    ------ --------- ----------- ----------- -----------
 master
    /remote/percy_dev/user1/sdc-DBs/master.dat
    file system device, special, dsync on, directio off, default disk,
    physical disk, 512.00 MB, Free: 351.00 MB
        3         0           0           0      262143
 private_dev1
    /percy_dev/user1/sdc-DBs/private_dev1.dat
    file system device, special, private 'cluster1_1', MIRROR DISABLED, non
    serial writes, dsync off, directio on, physical disk, 2.00 MB
        2050      0           1           0        1023
 regular_dev1
    /percy_dev/user1/sdc-DBs/dev1.dat
    file system device, special, dsync off, directio on, physical disk, 2.0
    0 MB
        2         0           2           0        1023
 tapedump1
    /dev/nst0
    unknown device type, disk, dump device
        16        2           0           0       20000
 tapedump2
    /dev/nst1
    unknown device type, tape,        625 MB, dump device
        16        3           0           0       20000(5 rows affected)
(return status = 0)
```

If `disk refit` is pending on a device, `sp_helpdevice` reflects it in its output.

In this example, "regular_dev" is a regular shareable device, while "private_dev1" and "private_dev2" are private devices owned by instance "cluster1_l". Further disk refit is pending on "private_dev1".

```
1> sp_helpdevice
2> go
 device_name
    physical_name
    description
    status cntrltype vdevno      vpn_low     vpn_high
 ----------
    ----------------------------------------------------------------------------
    ----------------------------------------------------------------------------
    ------ --------- ----------- ----------- -----------
 master
    /remote/percy_dev/user1/sdc-DBs/master.dat
    file system device, special, dsync on, directio off,
    default disk, physical disk, 512.00 MB, Free: 351.00 MB
        3         0           0           0      262143
 private_dev1
    /percy_dev/user1/sdc-DBs/private_dev1.dat
    file system device, special, private 'cluster1_1',
    REFIT PENDING, MIRROR DISABLED, nonserial writes, dsync off, directio
    on, physical disk, 2.00 MB
        2050       0           1           0        1023
 private_dev2
    /percy_dev/user1/sdc-DBs/private_dev2.dat         file system device,
special, private 'cluster1_1', MIRROR DISABLED, non
     serial writes, dsync off, directio on, physical disk, 2.00 MB
        2050       0           3           0        1023
 regular_dev1
    /percy_dev/user1/sdc-DBs/dev1.dat
    file system device, special, dsync off, directio on, physical disk, 2.00  MB
        2          0           2           0        1023
 tapedump1
    /dev/nst0
     unknown device type, disk, dump device
        16         2           0           0       20000
 tapedump2
    /dev/nst1
    unknown device type, tape,        625 MB, dump device
        16         3           0           0       20000(6 rows affected)
(return status = 0)
```

20.7.6 Using create database and alter database with a Private Device

Since a private device belongs to only one instance in the cluster, it can be used only for local user temporary databases on the instance that owns the private device.

When a private device is used to create or extend a database or its log, unless it is for a local user temporary database of the instance that owns the device, `create database` or `alter database` fails and an error message appears.

20.7.7 disk refit

If there are no private devices in the cluster, `disk refit` operates as usual.

If there are one or more private devices in the cluster, `disk refit` becomes a two-phased operation:

- Phase One performs `disk refit` only for those devices accessible to the instance that initiated the command – regular shareable devices and private devices owned by that instance.
- Phase Two performs `disk refit` only for the private devices of the instances in the cluster, one instance at a time. `disk refit` is complete only when both these phases are completed successfully.

20.7.7.1 Executing disk refit When There are Private Devices in the Cluster

Execute `disk refit` after restoring `sysdevices`.

Prerequisites

Restore the `sysdevices` table before you execute `disk refit`. Use the `disk reinit` command to add regular or private devices for any missing device entry in the `sysdevices` table.

For more information on restoring `sysdevices`, see the *SAP ASE System Administration Guide, Volume 2*.

Procedure

1. Restart the cluster in single-user mode by starting an instance in the cluster with the `-m` option. Single-user mode requires only one instance running in the cluster, and no other active connections to the cluster. Set trace flag 3608.
2. Issue the `disk refit` command on the instance. This removes all rows in `sysdatabases` and `sysusages`, and rebuilds parts of them from the devices in `sysdevices`, which are either regular sharable devices accessible by all instances in the cluster, or are private to this instance. At the end of Phase One, the cluster is shut down.

   SAP ASE prints messages at the end of Phase One that indicate whether `disk refit` is complete. If it is not, SAP ASE prints messages to inform the user which other instances in the cluster have private devices and must perform Phase Two.

   Alternatively, use `sp_refit_admin 'status'` to find instances with private devices on which `disk refit` has not yet run. A sample output:

   ```
   1> sp_refit_admin 'status'
   2> go
   You need to run DISK REFIT on the following instances to refit their private
   devices.
   ```

```
instance
device
status
----------------------------------------------------------------
cluster1_1
private_dev1
REFIT
PENDING
(1 row affected)
(return status = 0)
```

3. For each instance that has private devices and which still has to run `disk refit`:
   1. Start the instance by using the `-m` option and trace flag 3608.
   2. Issue `disk refit` on that instance.

   `disk refit` rebuilds `sysdatabases` and `sysusages` rows for private devices owned by this instance. When Phase Two for the instance completes, the cluster shuts down.
4. Even if you see no errors while executing `disk refit`, check the consistency of `sysdatabases` and `sysusages`.
5. Resume normal operation.

Related Information

Troubleshoot [page 235]

20.8 create database

Creates a new database.

> **Note**
>
> This information is specific to the Cluster Edition. For complete syntax and usage information for `create database`, see *SAP ASE Reference Manual: Commands*.

Syntax

```
create [ [ global | system ] temporary ] database <database_name>
    [ for instance <instance_name> ]
    [on {default | <database_device>} [= <size>]
        [, <database_device> [= <size>]]...]
    [log on <database_device> [= <size>]
        [, <database_device> [= <size>]]...]
    [with {override | default_location = "pathname"}]
    [for {load | proxy_update}]
```

Parameters

global temporary
: specifies that you are creating a global temporary database.

system temporary
: specifies that you are creating a local system temporary database.

temporary
: specifies that you are creating a temporary database.

`<database_name>`
: is the name of the new database. It must conform to the rules for identifiers and cannot be a variable.

for instance `<instance_name>`
: specifies the instance that is to own the local system temporary database or local temporary database you are creating. This parameter is not used when creating global temporary databases.

> **Note**
>
> Create a local temporary user database from the instance that is to own it. You can create a local system temporary database from any instance.

Examples

Example 1

While being in the owner instance ("ase1"), creates a local user temporary database on "ase1":

```
create temporary database local_tempdb1 for instance ase1
```

Alternatively, use:

```
create temporary database local_tempdb1
```

The `for instance <instance_name>` phrase is optional.

Example 2

Creates a local system temporary database on "ase1":

```
create system temporary database local)_systempdb1 for instance ase1
```

You can execute this command from any instance in the cluster.

Example 3

Creates a global temporary database:

```
create global temporary database global_tempdb1
```

20.9 sp_tempdb

`sp_tempdb` allows users to:

- Create temporary database groups
- Bind users or applications to the `default` or other temporary database group or to a specific local temporary database
- Manage bindings to local temporary databases and local temporary database groups

These bindings are stored in the `sysattributes` table in `master` database.

`sp_tempdb` provides the binding interface for maintaining bindings in `sysattributes` that are related to the multiple temporary database.

> **Note**
>
> This reference page contains information specific to the Cluster Edition. See the *SAP ASE Reference Manual* for complete information for `sp_tempdb`.

Syntax

```
sp_tempdb [
    [ { "create" | "drop" } , "<groupname>" ] |
    [ { "add" | "remove" } , "<tempdbname>", "<groupname>" ] |
    [ { "bind", "<objtype>", "<objname>", "<bindtype>", "<bindobj>"
        [, "<scope>", "<hardness>" ] } |
    [ { "unbind", "<objtype>", "<objname>"
        [, "scope" [, "<instance_name>",] ] } ] |
    [ "unbindall_db", "<tempdb_name>" ] |
    [ "unbindall_gr", <"groupname" >] |
    [ show [, "all" | "gr" | "db" | "login" | "app" [, "<name>" ] ] |
    [ who, "dbname"" ]
    [ help ] ]
```

Parameters

create

creates a temporary database group.

drop

drops a temporary database group.

<groupname>

is the name of the database group.

add

adds a temporary database to a temporary database group.

remove

removes a temporary database from a temporary database group.

<tempdbname>

is the name of the temporary database you are adding or removing.

bind

binds logins and applications to a temporary database or temporary database group.

unbind

unbinds logins and applications to a temporary database or database group.

<instance_name>

is the name of the instance owning the local temporary database that is to be unbound. (Cluster Edition only)

<objtype>

is the object type. Valid values are:

- `login_name` (or `LG`)
- `application_name` (or `AP`)

Values are not case-sensitive.

<objname>

is the name of the object you bind or unbind.

<bindtype>

is the bind type. Valid values are:

- `group` (or `GR`)
- `database` (or `DB`)

Values are not case-sensitive.

<bindobj>

is the name of the object being bound, and is either a group or a database depending on the `<bindtype>`.

<scope>

NULL.

<hardness>

is `hard`, `soft`, or NULL. The default is `soft`.

When you set the value of `<hardness>` to `hard`, a failure to assign a temporary database according to the binding results in a failure of the login. When you set the value to `soft`, such a failure results in the assignment of a temporary database from the default group or a local system temporary database.

unbindall_db

removes all login and application bindings for a given temporary database. It does not remove any database to group memberships. The `<tempdbname>` variable is required with this option.

Existing assignments to active sessions are not affected by this operation.

unbindall_gr **<groupname>**

removes all login and application bindings for a given temporary database group.

show

displays information stored in the `sysattributes` table about the existing groups, group members, login and application bindings, and active sessions that are assigned to a given database. The values are:

- `all` or no argument – displays the `default` temporary database group, all database-to-group memberships, and all login and application bindings.
- `gr` – displays the `default` temporary database group. `sp_tempdb show` displays all temporary databases bound to the `default` temporary database group whether you specify "default" for the `<name>` option or not.
- `db` – displays all databases to group memberships. If you provide `<name>`, then only the database to group memberships for the database `<name>` are printed.
- `login` – displays all login bindings where login is not NULL. If you provide `<name>`, then only the bindings for the login `<name>` are printed.
- `app` – displays all bindings where the application is not NULL. If you provide `<name>`, then the bindings for the application `<name>` are printed.

> **Note**
>
> `tempdb` is always part of the `default` database group.

who

displays all active sessions assigned to the given temporary database. When using the `who` parameter, use `<dbname>`.

Execute this command from the owner instance of the temporary `<dbname>`.

`<dbname>`

the name of a temporary database. If you provide a nontemporary database name for `<dbname>`, `sp_tempdb who` executes, but does not report any active sessions bound to it.

help

displays usage information. Executing `sp_tempdb` without specifying a command is the same as executing `sp_tempdb "help"`.

20.10 sys_tempdbid

Returns the ID of the local system temporary database of the specified instance.

Syntax

```
sys_tempdbid(spid)
```

Examples

Example 1

Returns the ID of the local system temporary database of the current instance:

```
select sys_tempdbid()
```

Example 2

Returns the ID of the specified local system temporary database:

```
select sys_tempdbid(3)
```

> **Note**
>
> To return the ID of a local system temporary database from an instance other than the owner instance, set the system view to "cluster".

Usage

- You can execute `sys_tempdbid` from any instance in the cluster.
- For general information about system functions, see the *SAP ASE Reference Manual: Building Blocks*.

Standards

ANSI SQL – Compliance level: Transact-SQL extension.

Permissions

Any user can execute `sys_tempdbid`.

20.11 tempdb_id

Reports the temporary database to which a given session is assigned. The input of the `tempdb_id` function is a server process ID (spid). Its output is the temporary database to which the process is assigned. If you do not provide a server process, `tempdb_id` reports the `dbid` of the temporary database assigned to the current process.

Syntax

```
tempdb_id(<spid>)
```

Examples

Example 1

Finds all the server processes that are assigned to a given temporary database:

```
select spid from master..sysprocesses
    where tempdb_id(spid) = db_id("tempdatabase")
```

> **Note**
>
> This example returns server processes for temporary databases owned by another instance only if the system view is set to cluster.

Usage

`select tempdb_id()` displays the same result as `select <@@tempdbid>`.

For general information about system functions, see the *SAP ASE Reference Manual: Building Blocks*.

21 Adding Space to an Archive Database

In general, access to an archive database is handled the same way in both clustered and nonclustered SAP ASE. When an archive database runs out of space, add space to the archive database using the `alter database` command.

In the SAP ASE Cluster Edition, run `alter database` from the same node that is updating the archive database. If you run `alter database` from a different node, SAP ASE prints an error message with the number of the node that is actually updating the archive database.

22 Memory in a Clustered Environment

The SAP ASE Cluster Edition requires more memory than a standalone nonclustered SAP ASE server. The additional memory supports internode messaging and cluster locking.

- You can configure the memory for internode messaging using the `CIPC regular message pool size` configuration parameter. The default value is 8 MB.
- To support distributed cache coherency, the SAP ASE Cluster Edition automatically configures locks for buffer caches and system descriptors when an instance starts.
  Each buffer in the data cache is automatically configured with a physical lock. The overhead for each physical lock is approximately 24.6% for databases with 2 KB pages and 12.3% for databases with 4 KB pages. Thus, for a 100 MB data cache, the additional memory overhead for physical locking is 24 MB for 2 KB pages and 12 MB for 4 KB pages.
  The overhead for system descriptors averages about 1.5 KB per open object. For best performance, the `lock hashtable size` parameter is automatically tuned to eight locks per bucket. For a large cache configuration, `lock hashtable size` may be automatically tuned to several megabytes.

23 Cluster Caches

Cluster cache configuration defines multiple named caches to have local or global caches according to application needs, which allows cluster instances to have local caches.

Objects can be bound to local or global caches. Multiple buffer pool support provides better access performance to named cache support by facilitating large I/Os.

You can also partition an application to localize access of application data to a particular instance that serves that application.

23.1 Global Caches

Global caches are defined for every instance in a cluster. For a global cache, attributes such as cache size, buffer pool setting, are specified in `sysconfigures` table, and all instances in a cluster read from this table to create the cache on their respective instances.

It is possible to change the attributes of a global cache such as cache size and buffer pool size to be instance-specific. If a particular instance has local settings, the cache is created using them. If the instance has no local definition, it uses a global definition to create the cache. In other words, instances that have local definitions override the global definitions and settings.

> **Note**
>
> You can increase the size of local and global caches dynamically, but you cannot reduce them dynamically.

23.2 Local Caches

Your application can define local caches for each instance in the cluster to cater to the instance's specific needs and to bind the cache to an object. A global definition is not required for each instance-specific cache in the cluster.

Local caches are instance-specific. You can tailor their configuration to the needs of the instance or the logical cluster to which the instance belongs. You should partition applications across instances within a cluster when a particular application usually runs in a particular instance. This maximizes the benefits of local caches, as you can configure them specifically for a particular application's access patterns.

An object can be bound to only one cache, either local or global, at any particular instance. If you do not bind the object to any cache, or in case of failover – where an instance-specific cache is not configured – it uses the default data cache. To aid efficient access, SAP ASE Cluster Edition maintains binding information on every instance.

> **Note**
>
> A local cache definition on any instance overrides any global definition at that instance.

23.3 Creating and Configuring Named Data Caches

Create and configure global and local data caches using `sp_cacheconfig`.

An SAP ASE installation contains a single global cache called the default data cache. The maximum data cache size is limited only by the amount of available memory in your system. The memory required to create the new cache is taken from the SAP ASE global memory. When the cache is created:

- It has a default wash size.
- The asynchronous prefetch size is set to the value of global asynchronous prefetch limit.
- It has only the default buffer pool.

The `sp_cacheconfig` system procedure creates and configures global and local data.

23.3.1 sp_cacheconfig

Creates, configures, reconfigures, and drops data caches, and provides information about them.

Syntax

```
sp_cacheconfig [<cachename> [, "<cache_size>[P | K | M | G]"]
    [, logonly | mixed ][, strict | relaxed]]
    [, "cache_partition=[1 | 2 | 4 | 8 | 16 | 32 | 64]"]
    [, instance <instance_name>]
```

Parameters

<cachename>

is the name of the data cache to be created or configured. Cache names must be unique, and can be up to 30 characters long. A cache name does not have to be a valid SAP ASE identifier, that is, it can contain spaces and other special characters.

<cache_size>

is the size of the data cache to be created or, if the cache already exists, the new size of the data cache. The minimum size of a cache is 256 times the logical page size of the server. Size units can be specified with `P` for pages, `K` for kilobytes, `M` for megabytes, or

G for gigabytes. The default is K. For megabytes and gigabytes, you can specify floating-point values. The cache size is in multiples of the logical page size.

logonly | mixed

specifies the type of cache.

strict | relaxed

specifies the cache replacement policy.

cache_partition

specifies the number of partitions to create in the cache.

<instance_name>

(In cluster environments) Is the name of the instance with a cache that you are adjusting.

Examples

Example 1

Creates the data cache `log_sales` that is 10MB in size and which is specific to instance `SALES_INSTANCE`. All space is in the default logical page size memory pool:

```
sp_cacheconfig log_sales, "10M",'instance SALES_INSTANCE'
go
```

If SAP ASE cannot allocate the amount of memory you request, it allocates all available memory and issues an error message that informs you of how much memory, in kilobytes, is being allocated dynamically when you restart the server.

This insufficient space may be due to unavailable memory or resource constraints. System administrators should make sure this condition is temporary, so that SAP ASE does not fail to restart in the future.

For example, if the maximum memory is 700MB and `tempdb_cache` is 100MB, making the server's total logical memory 600MB, and you attempt to add 100MB to `tempdb_cache`, the additional memory fits into maximum memory. However, if the server can allocate only 90MB, it allocates this amount dynamically, but the size field of the cache in the configuration file is updated to 100MB. On a subsequent restart, since SAP ASE obtains memory for all data caches at once, the size of `tempdb_cache` is 100MB.

By default, an `isql` connection has a cluster view. All instance-specific caches are displayed at any instance. For example, instance `HR_INSTANCE` displays information about cache `log_sales`, which is the instance-specific cache for `SALES_INSTANCE`". If you want `HR_INSTANCE` to see only the list of the local caches specific to this instance, and the global caches, set the system view to `instance`.

```
set system_view instance
go
```

Example 2

Displays all caches at instance `SALES_INSTANCE`, including both local and global configuration information::

```
sp_cacheconfig 'instance SALES_INSTANCE'
```

```
go
```

```
Cache Name                Status Type            Config Value   Run Value
------------------------- ------ -------------- ------------ ------------
default data cache Active Global,Default          0.00 Mb         8.00 Mb
SALES_INSTANCE:log_sales Active Mixed           100.00 Mb       100.00 Mb
                                            ------------ ------------
                                         Total 100.00 Mb 108.00 Mb
===============================================================================
Cache: default data cache, Status: Active, Type: Global,Default
    Config Size: 0.00 Mb, Run Size: 8.00 Mb
    Config Replacement: strict LRU, Run Replacement: strict LRU
    Config Partition: 1, Run Partition: 1
IO Size   Wash Size     Config Size   Run Size    APF Percent
-------- ------------- ------------ ------------ -----------
2 Kb       1638 Kb       0.00 Mb       8.00 Mb       10
===============================================================================
Cache: SALES_INSTANCE:log_sales, Status: Active, Type: Mixed
    Config Size: 100.00 Mb, Run Size: 100.00 Mb
    Config Replacement: strict LRU, Run Replacement: strict LRU
    Config Partition: 1, Run Partition: 1
IO Size      Wash Size  Config Size     Run Size   APF Percent
-------- ------------- ------------ ------------ -----------
2 Kb        20480 Kb       0.00 Mb    100.00 Mb        10
(return status = 0)
```

Example 3

Creates `tempdb_cache`, a 100 MB global cache in all instances, with memory allocated at all instances for the global cache:

```
sp_cacheconfig 'tempdb_cache', "100M"
go
```

Example 4

Creates a bigger named cache, `tempdb_cache`, at `SALES_INSTANCE`:

```
sp_cacheconfig 'tempdb_cache','150M','instance SALES_INSTANCE'
go
```

All cache operations can be executed from any instance.

> **Note**
>
> The local configuration of the named cache `tempdb_cache` overrides the global configuration. For example, if you set the system view to `cluster`, SAP ASE may display all the configurations for a named cache; ignore run values of any configuration that is not valid at that instance. For example, because `SALES_INSTANCE` has a valid local configuration, cache `tempdb_cache`, you should ignore the run values for global configuration. Similarly, since `HR_INSTANCE` has a valid global configuration, ignore the run values for local configuration of `tempdb_cache` that are related to `SALES_INSTANCE` at instance `HR_INSTANCE`.

Example 5

To add memory, use the syntax documented in the *SAP ASE Reference Manual: Procedures*. The additional memory you allocate is added to the SAP ASE page size pool. For example, the smallest size for a pool is 2K in a server with a logical page size of 2K. If the cache is partitioned, the additional memory is divided equally among the partitions.

> **Note**
>
> If adding memory to an existing global cache fails at an instance but succeeds at least one other instance, the server treats the operation as successful at the cluster-wide level. This makes it possible to have different run values for a global cache, but a single configuration value for the cache. `sp_cacheconfig` shows the run values of a global cache from the `syscurconfigs` entry of the current instance.

This example increases `tempdb_cache` size to 200MB, in instance `HR_INSTANCE`:

```
sp_cacheconfig 'tempdb_cache','200M'
```

Example 6

Decreases the size of `tempdb_cache` to 100M:

```
sp_cacheconfig 'tempdb_cache', '100M'
```

After you reduce a cache size, restart SAP ASE. After restarting SAP ASE, executing `sp_cacheconfig 'tempdb_cache'` displays:

```
Cache Name    Status Type        Config Value  Run Value
------------ ------ ------------ ------------ ------------
tempdb_cache Active Global,Mixed 100.00 Mb     100.00 Mb
        ------------ ------------
        Total 100.00 Mb 100.00 Mb
===============================================================================
Cache: tempdb_cache, Status: Active, Type: Global,Mixed
    Config Size: 100.00 Mb, Run Size: 100.00 Mb
    Config Replacement: strict LRU, Run Replacement: strict LRU
    Config Partition: 1, Run Partition: 1
IO Size    Wash Size    Config Size   Run Size     APF Percent
-------- ------------- ------------ ------------ -----------
2 Kb        20480 Kb      0.00 Mb     100.00 Mb       10
(return status = 0)
```

Example 7

Deletes a named cache `tempdb_cache` completely by reseting its size to 0:

```
sp_cacheconfig 'tempdb_cache', '0M', 'instance SALES_INSTANCE'
```

When you delete instance-specific configuration, a named cache reverts to its global configuration, if such a configuration exists.

If you try to delete a named cache that has objects bound to it, the server issues an error message.

- If the named cache has multiple configurations, the entry corresponding to the cache in the configuration file is deleted, as are the entries corresponding to the cache in `sysconfigures`. The cache is deleted the next time the instance is restarted.
- If the cache has a single configuration, either global or local, cache entry is not deleted from either the configuration file or from `sysconfigures`. This entry is deleted by either restarting the cluster or by creating a new configuration for the named cache.

Example 8

Creates a logonly cache called `log_hr` at `HR_INSTANCE`:

```
sp_cacheconfig 'log_hr','logonly','instance HR_INSTANCE'
```

Example 9

If a cache is dedicated to a table or an index, and the cache has little or no buffer replacement when the system reaches a stable state, you can set the relaxed LRU (least recently used) replacement policy.

The relaxed LRU replacement policy can improve performance for caches where there is little or no buffer replacement occurring, and for most log caches.

This example sets relaxed replacement policy for `log_sales` on `SALES_INSTANCE`:

```
sp_cacheconfig 'log_sales','relaxed','instance SALES_INSTANCE'
go
```

> **Note**
>
> Setting the cache replacement policy is not dynamic and requires you to restart the server.

For more information about `sp_cacheconfig`, see the SAP ASE *Reference Manual: Procedures*.

23.4 Configuring and Using Multiple Buffer Pools

Create multiple buffer pools using `sp_poolconfig`.

23.4.1 sp_poolconfig

Creates, drops, resizes, and provides information about memory pools within data caches.

Syntax

To create a memory pool in an existing cache, or to change pool size:

```
sp_poolconfig <cache_name>[, "<mem_size> [P | K | M | G]", "<config_pool>K"
    [, "<affected_pool> K"], instance <instance_name>]
```

To change a pool's wash size:

```
sp_poolconfig <cache_name>, "<affected_poolK>", "wash=<size>[P|K|M|G]"
```

To change a pool's asynchronous prefetch percentage:

```
sp_poolconfig <cache_name>, "<affected_poolK>",
    "local async prefetch limit=<percent> "
```

Parameters

<cache_name>

is the name of an existing data cache.

<mem_size>

is the size of the memory pool to be created or the new total size for an existing pool with the specified I/O size. The minimum size of a pool is 256 logical server pages. For a 2K logical page size server, the minimum size is 256K. Specify size units with `P` for pages, `K` for kilobytes, `M` for megabytes, or `G` for gigabytes. The default is kilobytes.

<config_pool>

is the I/O size performed in the memory pool where the memory is to be allocated or removed.

Valid I/O sizes are multiples of the logical page size, up to four times the amount.

<affected_pool>

is the size of I/O performed in the memory pool where the memory is to be deallocated, or the pools attributes such as `'wash size'` and `'prefetch limit'` are to be modified. If `<affected_pool>` is not specified, the memory is taken from the lowest logical page size memory pool.

<instance_name>

(Cluster Edition) is the name of the instance with the buffer pool you are adjusting.

wash=**<size>**

Changes the wash size (the point in the cache at which the SAP ASE server writes dirty pages to disk) for a memory pool.

local async prefetch limit=**<percent>**

sets the percentage of buffers in the pool that can be used to hold buffers that have been read into cache by asynchronous prefetch, but that have not yet been used. Valid values are 0–100. Setting the prefetch limit to 0 disables asynchronous prefetching in a pool.

Examples

Example 1

Creates a 4K pool for a named cache called `tempdb_cache` that is 25 MB in size:

```
sp_poolconfig 'tempdb_cache','25M','4K'
go
```

Example 2

Creates an 8K pool for named cache `log_hr`:

```
sp_poolconfig 'log_hr','50M','8K','instance HR_INSTANCE'
go
```

> **Note**
>
> There is no instance-specific configuration of buffer pools for a global cache. The instance option is used to detect the local cache configuration for which pool configuration is necessary.

Example 3

Creates a new 8K pool and take memory from a 4K pool rather than from a default pool, for named cache `log_hr`:

```
sp_poolconfig 'tempdb_cache','8M','8K','4K'
go
```

Example 4

Changes the wash size of an 8K pool of named cache `log_hr` to 12480K:

```
sp_poolconfig 'log_hr','8K','wash=12480K','instance HR_INSTANCE'
go
```

Example 4

Changes the local synchronous prefetch of named cache `log_sales` on instance `SALES_INSTANCE`:

```
sp_poolconfig 'log_sales','2K','local async prefetch limit=20','instance
SALES_INSTANCE'
go
```

> **Note**
>
> If a `dump database` command is active during a page split, the SAP ASE Cluster Edition issues a synchronous write on the new page the split produces.

Example 4

Drops a `4k` pool from named cache `tempdb_cache`, at instance `SALES_INSTANCE` by setting its size to 0:

```
sp_poolconfig 'tempdb_cache','0','4K'
go
```

> **Note**
>
> If a `dump database` command is active during a page split, the SAP ASE Cluster Edition issues a synchronous write on the new page the split produces.

For more information about `sp_poolconfig`, see the SAP ASE *Reference Manual: Procedures*.

23.5 Binding Objects to Named Caches

Use `sp_bindcache` system procedure to assign a database, table, index, text object, or image object to a cache.

To bind an entity to a cache:

- The named cache must exist with a status of "Active".
- The database or database object must exist.
- You can bind tables, indexes, or objects only from the database where they are stored.
- To bind system tables, including the transaction log table `syslogs`, the database must be in single-user mode.
- You can only bind a database from the `master` database.
- You can only bind a database, user table, index, text object, or image object to a cache of type "Mixed". Only the `syslogs` table can be bound to a cache of type "Log Only".
- To bind an object to a cache, you must either own the object or the database, or have system administrator status.

Binding objects to caches is dynamic. Cache binding or unbinding for local system temporary databases is not dynamic, and the owner-instance must be restarted for the bindings to take effect. Cache binding or unbinding for other temporary databases, including the global system temporary database, are dynamic.

23.5.1 sp_bindcache

Binds a database, table, index, `text` object, or `image` object to a data cache.

Syntax

```
sp_bindcache <cachename>, <dbname>
    [, [<ownername>.]<tablename>
    [, <indexname> | "text only"]]
```

Parameters

<cachename>

is the name of an active data cache.

<dbname>

is the name of the database to be bound to the cache or the name of the database containing the table, index, `text` or `image` object to be bound to the cache.

<ownername>

is the name of the table's owner. If the table is owned by "dbo", the owner name is optional.

<tablename>

is the name of the table to be bound to the cache, or the name of the table with an index, `text` object, or `image` object that is to be bound to the cache.

<indexname>

is the name of the index to be bound to the cache.

text only

binds `text` or `image` objects to a cache. When this parameter is used, you cannot give an index name at the same time.

Examples

Example 1

Binds the `SALES` database to a named cache `sales_cache`:

```
sp_bindcache 'sales_cache', 'SALES'
```

Usage

Cache binding is valid for the entire cluster. There is no instance-specific cache binding. If you bind an object to a local cache, the instance that has the cache configured for it uses that cache, and all other instances use the default data cache.

For complete documentation of `sp_bindcache`, see the *SAP ASE Reference Manual: Procedures*.

23.5.2 Getting Information About Bound Caches

Use `sp_helpcache` to obtain information about a cache and the entities bound to it.

The syntax is:

```
sp_helpcache {<cache_name> | "<cache_size>[P | K | M | G]" ,
    'instance <instance_name>'}
```

This example displays information about the `sales_cache` named cache:

```
sp_helpcache 'sales_cache'
go
```

```
Cache Name         Config Size Run Size
Overhead----------------- ----------- ---------- ----------
sales_cache:SALES_INSTANCE 50.00 Mb 50.00 Mb 6.39 Mb
```

```
(1 row affected)
--------------Cache Binding Information: --------
Cache Name Entity Name Type Index Name
Status
---------- ----------- ---- ----------
sales_cache SALES     database
V
(return status = 0)
```

For complete documentation of `sp_helpcache`, see the *SAP ASE Reference Manual: Procedures*.

23.5.3 sp_unbindcache and sp_bindcache_all

Unbinds a database, table, index, `text` object, or `image` object from a data cache.

Syntax

`sp_unbindcache` unbinds a single entity from a cache.

```
sp_unbindcache <dbname> [,[<owner>.]<tablename>
    [, <indexname >| "text only"]]
```

`sp_unbindcache_all` unbinds all objects bound to a cache.

```
sp_unbindcache_all <cache_name>
```

Parameters

<dbname>

is the name of database to be unbound or the name of the database containing the objects to be unbound.

<owner>

is the name of the table's owner. If the table is owned by the database owner, the owner name is optional.

<tablename>

is the name of the table to be unbound from a cache or the name of a table with an index, `text` object, or `image` object that is to be unbound from a cache.

<indexname>

is the name of an index to be unbound from a cache.

text only

unbinds `text` or `image` objects from a cache.

Examples

Example 1

Drops the cache binding of database `sales`:

```
sp_unbindcache 'SALES'
```

Usage

- When you drop a cache binding to an object, all the pages than are currently in memory are cleared from the cache.
- You cannot run `sp_unbindcache_all` on a named cache when there are system or remote local temporary databases bound to the cache. Use `sp_unbindcache` to unbind each of these databases before you run `sp_unbindcache_all`.

23.6 Configuration File

The named caches section in the configuration file accommodates instance information. The cache section of a global cache is similar to nonclustered SAP ASE output.

This is from the configuration file in a nonclustered SAP ASE environment:

```
[Named Cache:tempdb_cache]
    cache size = 100M
    cache status = mixed cache
    cache replacement policy = DEFAULT
    local cache partition number = DEFAULT
```

```
[2K I/O Buffer Pool]
    pool size = DEFAULT
    wash size = DEFAULT
    local async prefetch limit = DEFAULT
```

23.6.1 Format of a Local Named Cache

For the server to start, declare the global definition first in the configuration file, then declare the local definition. An instance-specific pool configuration on a named cache, for example, is not permitted if there is no corresponding instance-specific cache configuration.

This definition shows the format of a local named cache:

```
[Named Cache:log_sales]
    [Instance: SALES_INSTANCE]
```

```
    cache size = 100M
    cache status = mixed cache
    cache replacement policy = DEFAULT
    local cache partition number = DEFAULT
```

```
[2K I/O Buffer Pool]
    [Instance: SALES_INSTANCE]
    pool size = DEFAULT
    wash size = DEFAULT
    local async prefetch limit = DEFAULT
```

This example is not permitted:

```
[Named Cache:tempdb_cache]
    cache size = 100M
    cache status = mixed cache
    cache replacement policy = DEFAULT
    local cache partition number = DEFAULT
```

```
[2K I/O Buffer Pool]
    pool size = DEFAULT
    wash size = DEFAULT
    local async prefetch limit = DEFAULT
    [instance: SALES_INSTANCE]
    pool size = DEFAULT
    wash size = 40960K
    local async prefetch limit = DEFAULT
```

23.6.2 Extra Line in Local Cache Entries

Local caches and buffer pool definitions have an extra line (`[Instance: SALES_INSTANCE]`) that tells you that the configuration belongs to the instance "SALES_INSTANCE".

If a named cache has both global and local configurations, the cache section of the configuration file shows:

```
[Named Cache:tempdb_cache]
    cache size = 100M
    cache status = mixed cache
    cache replacement policy = DEFAULT
    local cache partition number = DEFAULT
    [Instance: SALES_INSTANCE]
    cache size = 150M
    cache status = mixed cache
    cache replacement policy = DEFAULT
    local cache partition number = DEFAUL
```

```
[2K I/O Buffer Pool]
    pool size = DEFAULT
    wash size = DEFAULT
    local async prefetch limit = DEFAULT
    [Instance: SALES_INSTANCE]
    pool size = DEFAULT
    wash size = 40960k
    local async prefetch limit = DEFAULT
```

23.6.3 Deleted Named Cache with Global Configuration

When a named cache is deleted, the configuration file includes `cache status = deleted`.

The cache configuration entry looks similar to:

```
[Named Cache: tempdb_cache]
    cache size = 100M
    cache status = deleted
    cache replacement policy = DEFAULT
    local cache partition number = DEFAULT
```

```
[2K I/O Buffer Pool]
    pool size = DEFAULT
    wash size = DEFAULT
    local async prefetch limit = DEFAULT
```

23.6.4 Named Cache with Local Configuration

If the named cache is configured locally, the cache section entry looks similar to:

```
[Named Cache: tempdb_cache]
    [Instance: SALES_INSTANCE]
    cache size = 100M
    cache status = deleted
    cache replacement policy = DEFAULT
    local cache partition number = DEFAULT
```

```
[2K I/O Buffer Pool]
    [Instance: SALES_INSTANCE]
    pool size = DEFAULT
    wash size = DEFAULT
    local async prefetch limit = DEFAULT
```

23.6.5 Deleted Entries with Valid Configuration

The configuration file should not contain any deleted entries that have at least one valid configuration.

For example:

```
[Named Cache: tempdb_cache]
    cache size = 100M
    cache status = mixed cache
    cache replacement policy = DEFAULT
    local cache partition number = DEFAULT
```

```
[Instance: SALES_INSTANCE]
    cache size = 150M
    cache status = deleted
    cache replacement policy = DEFAULT
```

```
    local cache partition number = DEFAULT
```

```
[2K I/O Buffer Pool]
    pool size = DEFAULT
    wash size = DEFAULT
    local async prefetch limit = DEFAULT
    [Instance: SALES_INSTANCE]
    pool size = DEFAULT
    wash size = 40960K
    local async prefetch limit = DEFAULT
```

23.6.6 Consequences of Creating a Local Configuration in the Presence of a Global Configuration

When you create a local configuration of a named cache in the presence of global configurations, all pool entries are replicated for local configuration.

For example, you may have the following global configuration:

```
[Named Cache: tempdb_cache]
    cache size = 100M
    cache status = mixed cache
    cache replacement policy = DEFAULT
    local cache partition number = DEFAULT
```

```
[2K I/O Buffer Pool]
    pool size = DEFAULT
    wash size = DEFAULT
    local async prefetch limit = DEFAULT
```

```
[4K I/O Buffer Pool]
    pool size = 25.0000m
    wash size = DEFAULT
    local async prefetch limit = DEFAULT
```

If you create a local configuration with size 120 MB on this global configuration, the cache section of the configuration file looks similar to:

```
[Named Cache:tempdb_cache]
    cache size = 100M
    cache status = mixed cache
    cache replacement policy = DEFAULT
    local cache partition number = DEFAULT
    [Instance:SALES_INSTANCE]
    cache size = 120M
    cache status = mixed cache
    cache replacement policy = DEFAULT
    local cache partition number = DEFAULT
```

```
[2K I/O Buffer Pool]
    pool size = DEFAULT
    wash size = DEFAULT
    local async prefetch limit = DEFAULT
    [Instance:SALES_INSTANCE]
    pool size = DEFAULT
    wash size = DEFAULT
    local async prefetch limit = DEFAULT[4K I/O Buffer Pool]
```

```
    pool size = 25.0000m
    wash size = DEFAULT
    local async prefetch limit = DEFAULT
    [Instance:SALES_INSTANCE]
    pool size = 25.0000m
    wash size = DEFAULT
    local async prefetch limit = DEFAULT
```

23.7 Limitations

If a named cache has only local configurations and objects bound to it, you cannot create a global cache configuration, and SAP ASE Cluster Edition displays an error message.

For example:

```
sp_cacheconfig 'SALES', '200M', 'instance SALES_INSTANCE'
go
sp_bindcache 'SALES','SALES_DB'
go
sp_cacheconfig 'SALES', '300m'
go
-------
Error 19817, "New configuration on cache 'SALES' is not permitted."
```

To avoid such a situation:

1. Unbind all objects bound to cache `SALES`.
2. Create a new configuration for cache `SALES`.
3. Bind the objects again.

To avoid physical lock deadlocks, the index pages in the SAP ASE Cluster Edition do not use large pool support.

24 Locks in a Clustered Environment

All database data and metadata objects can be accessed and cached by any instance in the cluster. For this reason, these objects are subject to distributed locking and cache coherency enforcement, as are runtime global data structures and variables.

The cluster lock manager (CLM) provides distributed locking services for sharing globally accessible and cacheable objects. The CLM creates and queues locks on behalf of locking requests. It arbitrates locking requests for access rights from any instance to any global object in the shared-disk cluster environment.

Locks can have task, transaction, or instance ownership. Locks managed by the local lock manager have task or transaction ownership. Locks managed by the CLM have instance ownership; they are shared among all processes or transactions on that instance, and are retained until another instance requests a conflicting lock mode.

24.1 Deadlocks

For all-pages locking in a nonclustered SAP ASE, if the server fails to acquire a page lock, it retries the attempt instead of waiting for the lock (waiting could cause a deadlock). The `deadlock retries` configuration parameter determines the number of retries that a server attempts.

The SAP ASE Cluster Edition includes a physical lock, which ensures page consistency across nodes. However, a physical lock is not included in the deadlock detection mechanism, so a server cannot detect a deadlock.

The SAP ASE Cluster Edition retries failed attempts to acquire a lock. However, when the server exceeds the value for `deadlock retries`, it fails or rolls back the query instead of waiting on the lock and risking an undetected deadlock.

24.2 Retention Locks

All SAP ASE Cluster Edition locks are retention locks, which are cluster-wide locks granted to instances in the cluster, with ownership shared among all processes on the instances.

Ownership of a retention lock is retained until another instance claims a conflicting lock mode, or the resource associated with the lock is claimed by another instance. The use of retention locks reduces the need to acquire and release locks, and thus reduces intra-instance messaging.

The SAP ASE Cluster Edition uses several different retention locks.

24.3 Object Coherency for Metadata and Global Variables

In the cluster environment, all instances share in-memory metadata and global variables. Object coherency requires the ability to serialize updates to these shared objects and make the latest changes available to all instances in the cluster.

The strategies for maintaining object consistency vary depending on the type of data structure, its usage, and the type of locking protocol. There are two types of coherency:

- Data coherency – requires two or more copies of the same data structure to be consistent. It generally implies "strict consistency", where the copies are identical. Sometimes "loose consistency" is sufficient. In these cases, the data structures are related according to certain constraints.
- Behavioral coherency – requires instances to perform consistent operations, so that one instance cannot drop an object and, at the same time, allow another instance to access that object.

To establish object coherency, SAP ASE Cluster Edition makes use of the following:

- Cluster-wide locking – establishes ownership of an object, enables older versions to be invalidated, and permits transfer of the latest version on demand. Cluster-wide locking is typically used when strict coherency is required. See "Locking" for a description of the locks used in SDC.
- Messaging – requests a specific data item or operation, aggregates local data (such as keep count), and updates statistics and counters. Messaging is usually reserved for objects requiring only loose consistency or behavioral consistency.

Whenever possible, the system uses cluster-wide locking to maintain object coherency, because cluster-wide locking is more efficient than intra-instance messaging.

24.4 Cluster Lock Requests and Task Request Status

In the SAP ASE Cluster Edition, a task lock request triggers the cluster lock request. Once the cluster lock request is made, however, the cluster lock request is independent of task request status.

Suppose, for example, the task requesting the lock leaves before the lock is granted at the cluster level. This could happen if the lock times out or the task is killed. In such a case, the output of `sp_lock`, for the instance owning the cluster lock, continues to show `-blk` – indicating a blocking process, even though there may be no task waiting for the lock – until the instance owning the cluster lock releases it.

24.5 Enabling Lock Remastering

SAP ASE includes the `enable lock remastering` configuration parameter, which allows cluster server processes to manage locks locally once failed-over client connections migrate back to the original instance.

In versions earlier than 15.7 SP130, if the client connections moved to an undesignated instance, SAP ASE allocated new lock masters on the undesignated instance, even though the client connections are later moved back to the designated instance. Because the lock masters in these versions are remote, any connections requiring these locks had to travel to the remote master to be granted locks.

For SAP ASE versions 15.7 SP130 and later, lock masters are allocated when clients connect to an instance, and remain on the current instances until garbage collection occurs. Once you enable the `enable lock remastering` configuration parameter, SAP ASE detects the lock masters allocated on the undesignated instances, and removes them from the undesignated instances after the client connections are moved back.

The `monCLMInfo` monitoring table includes these columns, which track locking activity:

- `RatioLMUGCLockSkip`
- `AvgLMUGCTime`
- `MaxLMUGCTime`
- `RatioPMUGCLockSkip`
- `AvgPMUGCTime`
- `MaxPMUGCTime`

24.6 Preventing Out-of-Lock Situations

`physical lock cushion` allows you to allocate a sufficient number of locks to prevent out-of-lock conditions, which can occur during instance startup, shutdown, failover, or on badly partitioned environments.

The value for `physical lock cushion` is a percentage, and can be any value between 0 (the default) and 100.

The number of physical locks needed is directly proportional to the cache pool size. The larger the pool size, the more physical locks SAP ASE allocates.

Typical situations that result in SAP ASE running out of locks are:

- Creating badly partitioned clusters that include instances that access the same data. SAP ASE can run out of locks when badly partitioned tables are frequently accessed. This can cause SAP ASE to create additional locks with lock records that consume more lock structures than well-partitioned table queries.
- A delayed instance startup, which can result in the cluster hosting all master locks on a single node. If you start an instance and its cache is filled, it retains too many master locks to support additional instances started later that may access the same pages.

Set `physical lock cushion` to 100 (particularly for badly partitioned clusters for which you have configured data caches) to fully prevent SAP ASE from running out of physical locks. In this case, SAP ASE requires additional memory (which is proportional to the total data cache size) to allocate the additional physical locks.

The number of physical locks you require depends on:

- The total data cache size (C)
- The data page size (P)
- The cluster redundancy level (CRL)
- The number of nodes (N).

The larger the value for the total data cache size, cluster redundancy level, or the number of nodes, the more physical locks SAP ASE requires. The size of each lock is 184 bytes.

The formula for the number of physical locks is:

```
(C/P)X(CRL+2)
```

The formula for the maximum number of physical locks needed to prevent out of lock conditions is:

```
(C/P)X(MAX(CRL+3, N+1))
```

The number of additional locks needed is determined by the difference between these two formulas:

```
(C/P)X(MAX(CRL+3,N+1)-(CRL+2))
```

The formula to determine the amount of additional memory needed is

```
(C/P)X(MAX(CRL+3,N+1)-(CRL+2))X 184 X (physical_lock_cushion)
```

For example, a cluster with this configuration:

- Total cache size (C) = 1GB
- Data page size (P) = 4KB
- Cluster redundancy level (CRL) = 1
- Number of nodes (N) = 4

Creates:

- 262144 buffers – 1GB/4KB
- 786432 physical locks – 262144 x 3
- 524288 additional locks – 262144 x 2
- 100M of additional memory – 262144 x 2 x 184

SAP ASE uses the physical lock cushion to allocate the percentage of the maximum locks needed. In a well-partitioned environment, tables are accessed from a single instance, which reduces the number of physical locks needed.

No additional physical locks are needed for a two-node cluster in a well-partitioned active-passive environment (the physical lock cushion can be zero). No additional physical locks are needed for:

- 3-node clusters in a well-partitioned active-passive environment where CRL = 2
- 4-node cluster in a well-partitioned active-passive environment where CRL = 3
- A cluster in a well-partitioned active-passive environment where CRL = N - 1

The physical lock cushion for all these configurations can be 0.

24.7 Lock Management Enhancements

SAP ASE 16.0 SP01 Cluster Edition introduces several enhancements to lock management in the following areas: engine lock transfer, engine lock caching, lock promotion, and deadlock checking.

See the *Performance and Tuning: Locking and Concurrency Control>Locking Configuration and Tuning>Locks and Performance>Reduce Lock Contention>Lock Manager Enhancements*.

> **Note**
>
> The optimizations in the areas of `LOCK_VERIFY` operations, log semaphore locks, and hot DOL tables are not available in a clustered environment.

24.8 Improved Concurrency for Partition-Level Online Operations

Certain partition-level operations can concurrently operate on different partitions of a table. DML can also concurrently operate on the table while the partition level online operation is running.

See the *Performance and Tuning: Locking and Concurrency Control>Locking Configuration and Tuning>Locks and Performance>Reduce Lock Contention>Lock Manager Enhancements*.

> **Note**
>
> In a clustered environment, multiple concurrent partition-level utilities on different partitions of the same table should be executed from one node. If partition-level operations are initiated from two or more different nodes, only one of them will succeed while others will fail with error 9859 or 8233.

25 Thresholds in a Clustered Environment

A user-created stored procedure executes when free space on a segment drops below the threshold maintained by SAP ASE.

All databases have a last-chance threshold on the log segment, which is an estimate of the number of free log pages that are required to back up and truncate the transaction log. The last-chance threshold (LCT) is a defined amount of available space; a specified number of free pages on a log segment.

The LCT also defines the action to be taken when the amount of space falls below the specified threshold value. This threshold monitors free space within a database, preventing the transaction log from running out of space.

When you add a threshold to a segment, specify the stored procedure used to monitor this threshold. By default, SAP ASE uses `sp_thresholdaction` for the last-chance threshold.

In general, threshold maintenance in cluster processing is indistinguishable from that of a nonclustered SAP ASE. However, there are some differences in the following `dbcc` commands:

- `dbcc thresholds`
- `dbcc dbtable`
- `dbcc dbrepair` with `remap`
- `dbcc dbrepair` with `newthreshold`

25.1 dbcc thresholds Output

`dbcc thresholds` prints segment structure.

The SAP ASE Cluster Edition output differs because segment the structure itself is changed. For example, the `sg_below` and `sg_above` threshold pointers for the segment are numbers that specify the positions of the two thresholds in the `dbt_thresholds` array.

25.2 dbcc dbtable Output

`dbcc dbtable` prints the `dbt_thrmgr_info`, which contains cluster threshold management data, and prints the segment structures in the table.

25.3 dbcc dbrepair with remap Option

`dbcc dbrepair` with `remap` performs functions related to segments and disk maps.

- Rebuilds the disk map in `dbtable`
- Recalculates the segment free page counts
- Sets up the thresholds in each segment according to the threshold levels on the segment and the total free page count

Use `dbcc dbrepair` with `remap` to specify either the segment number or segment name, but you can specify only one segment at a time to remap.

If you do not specify the segment to remap, SAP ASE remaps all the segments in the database. You can use the optional parameters at the end of `dbcc dbrepair` to specify the segment for remapping.

`dbcc dbrepair` includes the option `fixalloc`; to use both `remap` and `fixalloc`, add `remap` before `fixalloc`. Specify the segment number or segment name at the end of this command. This example specifies the log segment for the `pubs2` database:

```
dbcc dbrepair(pubs2,"remap","fixalloc", -1,"logsegment")
```

This example remaps the log segment in the `pubs2` database without using `fixalloc`:

```
dbcc dbrepair(pubs2, "remap",NULL, -1,"logsegment")
```

This example remaps all segments in the `pubs2` database:

```
dbcc dbrepair(pubs2, "remap")
```

The `dbcc dbrepair` with the `remap` option is used by `sp_addsegment`, `sp_dropsegment`, `sp_modifysegment`, `sp_extendsegment`, `sp_logdevice`, and `sp_placeobject`.

For more information on `dbcc`, see the SAP ASE *Reference Manual: Commands*. For more information on the system procedures, see the SAP ASE *Reference Manual: Procedures*.

25.4 dbcc dbrepair with newthreshold Option

`dbcc dbrepair` with `newthreshold` performs functions related to thresholds.

- Loads thresholds in a database, from `systhresholds` to `dbt_thresholds` in `dbtable`.
- Sets up the threshold pointers in each segment according to the levels of the thresholds on the segment, and the segment-free page count.
- Specifies the segment name or number, and reads all thresholds in `systhresholds` into `dbt_thresholds`. Only the thresholds in `dbt_thresholds` segment are set up according to the thresholds that belong to the `dbt_thresholds` segment and the segment-free page count.

The `dbcc dbrepair` with `newthreshold` is used by `sp_addthreshold`, `sp_dropthreshold`, and `sp_modifythreshold`.

For more information on `dbcc`, see the SAP ASE *Reference Manual: Commands*. For more information on the system procedures, see the SAP ASE *Reference Manual: Procedures*.

26 Cluster Inter-process Communication

Cluster inter-process communication (CIPC) is a critical subsystem in the cluster architecture. Most cluster subsystems use CIPC to communicate with other instances in the cluster.

> **Note**
>
> A high-speed CIPC network is necessary for an acceptable SAP ASE Cluster Edition performance.

CIPC supports UDP and RDMA protocols for communication.

Tasks running on instances within the cluster use CIPC to communicate with other tasks in other instances in the cluster. CIPC information specified in the cluster input file and stored on the quorum device is used to connect all instances in the cluster. Within the input file, CIPC supports a primary and an optional secondary interconnect. The secondary interconnection, if present, is used for buffer-cache transfers, reducing the load on the primary interconnection.

> **Note**
>
> Secondary interconnected networks are not supported with RDMA.

26.1 Remote Directory Memory Access (RDMA)

The remote direct memory access (RDMA) transfers data directly from the virtual memory map of one server into that of another without the involvement of the operating system. This permits high-throughput, low-latency networking, which is especially useful in massively parallel computer clusters.

> **Note**
>
> RDMA requires InfiniBand and supports only one primary interconnect. The primary address for each server in the cluster must correspond to an InfiniBand channel adapter. See *The Cluster Input File* in the *SAP Adaptive Server Enterprise Installation Guide*.

RDMA direct memory transfers require no work by CPUs and no context switches, and transfers continue in parallel with other system operations. When an application performs an RDMA send or receive request, the application data is accessed and transferred directly by the channel adapter, reducing latency and enabling faster message transfer.

> **Note**
>
> With RDMA, some CIPC messages are reserved for pre-posting of messages for receive requests. When RDMA is enabled, avoid running out of CIPC messages by increasing these default CIPC message configurations by about 50%:
>
> - `CIPC regular message pool size`

- `CIPC large message pool size`

See *System Administration Guide: Volume 1 > Setting Configuration Parameters > Configuration Parameters > Alphabetical Listing of Configuration Parameters*.

Related Information

Changing Cluster Configuration Values [page 36]

27 Using InfiniBand

SAP ASE supports InfiniBand (IPoIB) for internal communication between nodes in a cluster.

To use InfiniBand:

- Configure the host channel adaptor (HCA).
- Use an InfiniBand software stack.
- For Linux, the SAP ASE Cluster Edition requires OFED 1.2 from OpenFabrics.

27.1 Setting the Buffer Space

The SAP ASE Cluster Edition rquires sufficient buffering space to ensure adequate performance, particularly on faster interconnects. The default values for buffer space on Linux and Solaris systems are too small for gigabit Ethernet and InfiniBand. Modify them to ensure adequate networking performance.

27.1.1 Configuring Buffer Space on Linux

Set the buffer space to an appropriate size on your system using a command similar to the following:

```
/sbin/sysctl -w net.core.rmem_max=<value>
/sbin/sysctl -w net.core.wmem_max=<value>
```

On most platforms, the default value for `rmem_max` is set to about 128KB, which is too small for the SAP ASE Cluster Edition. Increase `rmem_max` to the largest value your platform allows (1 MB is a good starting value):

```
/sbin/sysctl -w net.core.rmem_max=1048576
```

27.2 Configuring InfiniBand in a Cluster

After you configure the host channel adapter (HCA), the `/etc/hosts` file includes a corresponding host name and IP address.

Use this host name or IP address when SAP Control Center or `sybcluster` asks for this information during the cluster configuration. If you manually configure the cluster, add the IP address.

28 Support for mount and unmount Commands

You can use the `mount database` and `unmount database` commands in a shared-disk cluster.

If an instance fails while one of these commands is in progress, the command may abort. Reissue `mount database` or `unmount database` when the instance failover recovery is complete.

29 Using sp_showplan

Do not use `sp_showplan` across multiple nodes of a shared-disk cluster. It is specific to a single connection to a node

30 Restrictions for Using Encrypted Columns in a Clustered Environment

In a clustered environment, there are restrictions for using encrypted columns.

You cannot:

- Create encryption keys in local temporary databases
- Set the system encryption password in local temporary databases

See the *SAP ASE Encrypted Columns Users Guide*.

31 Adding SSL Common Name

The SAP ASE Cluster Edition allows the server name specified in the directory service entry to differ from the common name that the Secure Sockets Layer (SSL) server certificate uses to perform an SSL handshake.

Context

This allows you to use a fully qualified domain name for the SSL certificate common name (for example, `server1.bigcompany.com`) and use the same certificate for multiple servers.

Procedure

Add a common name to the interfaces file using this format, where `<common_name>` is the fully qualified domain name for the cluster node:

```
ase1
   master tcp ether <host_name> <port_number> ssl="CN='<common_name>'"
   query tcp ether <host_name> <port_number> ssl="CN='<common_name>'"
ase2
   master tcp ether <host_name> <port_number> ssl="CN='<common_name>'"
   query tcp ether <host_name> <port_number> ssl="CN='<common_name>'"
ase3
   master tcp ether <host_name> <port_number> ssl="CN='<common_name>'"
   query tcp ether <host_name> <port_number> ssl="CN='<common_name>'"
mycluster
   query tcp ether <host_name> <port_number> ssl="CN='<common_name>'"
   query tcp ether <host_name> <port_number> ssl="CN='<common_name>'"
   query tcp ether <host_name> <port_number> ssl="CN='<common_name>'"
```

`<common_name>` can include white space. Instances defined in the interfaces file may or may not use the same common name.

> **Note**
>
> You can add only one SSL certificate to a `master` database. Because each instance in a cluster shares the same disk, they all use the same path for the SSL server certificate. We recommend that all instances use the same common name.

This is a sample interfaces file entry for cluster mycluster:

```
ase1
   master tcp ether blade1 19786 ssl="CN='ase1.big server 1.com'"
   query tcp ether blade1 19786 ssl="CN='ase1.big server 1.com'"
ase2
   master tcp ether blade2 19886 ssl="CN='ase1.big server 1.com'"
   query tcp ether blade2 19886 ssl="CN='ase1.big server 1.com'"
ase3
```

```
   master tcp ether blade3 19986 ssl="CN='ase1.big server 1.com'"
   query tcp ether blade3 19986 ssl="CN='ase1.big server 1.com'"
mycluster
   query tcp ether blade1 19786 ssl="CN='ase1.big server 1.com'"
   query tcp ether blade2 19886 ssl="CN='ase1.big server 1.com'"
   query tcp ether blade3 19986 ssl=ssl="CN='ase1.big server 1.com'"
```

31.1 Specifying a Common Name with sp_listener

Use `sp_listener` to specify a common name for an SSL certificate. `sp_listener` includes the `CN=<common_name>` parameter, which allows you to specify a common name for the SSL certificate.

The syntax is:

```
sp_listener 'command','[protocol:]<machine_name>:<port_number>:
"CN=<common_name>"', '<engine_number>'
```

Use `CN=<common_name>` only when you specify `ssltcp` as the protocol. If included, it uses the specified `<common_name>` to validate the `<common_name>` in the SSL certificate. If you do not include `CN=<common_name>`, SAP ASE uses `<server_name>` to validate the common name in the SSL certificate. `CN=<common_name>` must match the common name entry in the certificate. Any fully qualified domain name in the certificate must match `CN=<common_name>`.

The attribute name "CN" is case-insensitive, but the attribute value for the common name is case-sensitive. For example, the attribute name may be "CN", "Cn", or "cn".

This example specifies the common name `ase1.big server 1.com`:

```
sp_listener 'start','ssltcp:blade1:17251:"CN=ase1.big server 1.com"','0'
```

See the *SAP ASE Reference Manual: Procedures* for more information about `sp_listener`.

32 Removing Column Required for the Search Argument from the Substitution List

SAP ASE removes columns from a query's substitution list if they are required only for the search argument (and therefore need not be decompressed).

This example query includes columns `c1`, `c2`, `c3`, and `c4` from a compressed table `t1`, all in the substitution list:

```
select c1 from t1 where c2 = 1 and c3 > 1 and c4 < 100
```

The query processor includes columns `c2`, `c3`, and `c4` in the search argument, but removes them from the decompression list.

Use `dccc traceon(550)` to view the substitution list for each table.

33 Performing Lazy Decompression Using a RID Join

SAP ASE includes a `RIDJoin` operator above the `nested loop join` operator in the query plan to compress only the necessary columns during a nested loop join.

This approach is known as lazy decompression, and delays decompressing the row or columns involved as late as possible in the query plan execution. This example illustrates a typical nested loop join between table `t3` (compressed) and table `t4` (uncompressed):

```
select t3.* from t3, t4
where t3.c1=t4.c1
                           Emit
                            /
                           /
                       RIDJoin
                      /       \
                     /         \
             NestLoopJoin    TableScan
              Inner Join        t3
              /       \
             /         \
      TableScan     TableScan
         t3           t4
```

Instead of decompressing all of the columns and then joining them with table `t4` to find a match for column `c1`, the `RIDJoin` operator decompresses only `t3.c1` and compares it with `t4.c1` during the nested loop join for each row in table `t3`.

Lazy decompression includes these limitations:

- You cannot include multiple nested `RIDJoin` operators. If two compressed tables join, the query processor selects the larger table for the lazy decompression for a hash join, and chooses the left table for the lazy decompression for a nested loop join.
- You cannot include merge join and subquery cases.
- Since SAP ASE compression depends on the data involved, you cannot perform lazy decompression on tables that have all data compressed on all rows. However, because the query processor decides on lazy decompression during the query plan compilation stage, it uses lazy decompression as long as the table has the compress status enabled.
- It is not caused by the SAP ASE optimizer.

34 Configure the Operating System

You can adjust the operating system configuration settings after installing or upgrading the SAP ASE Cluster Edition. Unless stated otherwise, the information pertains to all supported UNIX platforms.

34.1 Using the stty Setting

Set `stty tostop` to stop a background SAP ASE from writing output to the terminal.

Procedure

To stop background output, enter this command before starting SAP ASE:

```
stty -tostop
```

You need not change the `stty` setting if if you are redirecting all SAP ASE output to files.

34.2 Restoring Correct Permissions

SAP ASE-related software files and directories are installed with the correct access permissions. If necessary, you can restore the correct permissions with the script `setperm_all`, located in `$SYBASE/$SYBASE_ASE/install`.

34.3 File Descriptors and User Connections

Find or change the current file descriptor values. The number of user connections used by SAP ASE cannot exceed the number of file descriptors available to SAP ASE on the operating system.

When configuring user connections on SAP ASE, the system administrator should take into account the number of file descriptors available per process. Although most of the open file descriptors are available for user connections, a few are used by SAP ASE to open files and devices.

Platform	Description
Linux	The number of file descriptors per process is limited to 10,000. You can set the number of file descriptors using `ulimit`. SAP ASE can use a maximum of 40000 file descriptors.

34.3.1 Displaying Current Soft and Hard Limits

Use `limit` and `unlimit` commands to display the current soft and hard limits.

- Display the current soft limit:
  - C shell:

    ```
    limit descriptors
    ```

  - Bourne shell:

    ```
    ulimit -n
    ```

- Display the current hard limit:
  - C shell:

    ```
    limit -h descriptors
    ```

  - Bourne shell:

    ```
    ulimit -Hn
    ```

34.3.2 Increasing the Soft Limit

Increase the soft limits for C and Bourne shells.

To increase the soft limit, enter:

- (C shell)

  ```
  limit descriptors <n>
  ```

- (Bourne shell)

  ```
  ulimit -Sn <new_value>
  ```

where:

- `<n>` is the current value for the soft limit.
- `<new_value>` is the value to which to increase the soft limit.

> **Note**
>
> You can use the preceding commands in your `RUN_server_name` Bourne shell script to increase the hard and soft limits.

34.3.3 Increasing the Hard Limit

Set up a program to increase the hard limit.

Procedure

1. Create `file_name.c`, where `file_name` is the name you give the file.
2. Enter the text shown below, making the appropriate modifications for your situation. .

   > **Note**
   >
   > This is an sample script; modify it as necessary.

```
#include <sys/time.h>
 #include <sys/resource.h>
 #include <sys/types.h>
 /*
 ** define MAX_CONNECTIONS to a number less than
 ** 10000. The number defined will then become the maximum
 ** number of connections allowed by an SAP ASE.
 */
 #define MAX_CONNECTIONS 9999
 extern int errno;

 main(argc,argv)
 char **argv;
 {
     struct rlimit rlp;
     uid_t uid;

     rlp.rlim_cur = MAX_CONNECTIONS;
     rlp.rlim_max = MAX_CONNECTIONS;
  /* set the number of open file descriptors to
     MAX_CONNECTIONS */
     if (setrlimit (RLIMIT_NOFILE,&rlp) == -1)
     {
        perror("setrlimit");
        exit(1);
     }

   /* reset the user id to disable superuser
     privileges */
     uid = getuid();
     setuid(uid);
   /* run the program indicated as arguments to
     this program */
     execv(*++argv, argv);
 }
```

3. Compile the file:

```
cc <file_name.c> -o <program_name>
```

   where:
   - `<file_name.c>` is the source file that you created.
   - `<program_name>` is the compiled file name.
4. Change the program's permissions and ownership:

```
chmod 755 <program_name>
chown root <program_name>
```

5. The root user can start SAP ASE with increased user connections by entering this command at the operating system prompt.

```
# <program_name> dataserver -d <master_device_name>
```

   where `<master_device_name>` is the full path of the SAP ASE master device. Alternatively, you can add `<program_name>` preceding the `dataserver` command line in the SAP ASE `RUN_server_name` file.

   See *Setting Configuration Parameters* in the *System Administration Guide, Volume 1*.

34.4 Adjust the Client Connection Timeout Period

SAP ASE uses the `KEEPALIVE` option of the TCP/IP protocol to detect inactive clients.

When a connection to a client is inactive for a period of time (the *timeout period*), the operating system sends `KEEPALIVE` packets at regular intervals. If it does not receive a response from the client machine for any of these packets, the operating system notifies SAP ASE that the client is no longer responding. SAP ASE then terminates the client's connection.

The `KEEPALIVE` default timeout period is 2 hours (7,200,000 ms). To display the current time value, use the appropriate command for your platform.

34.4.1 Adjusting the Client Connection Timeout

Display or change the current timeout period.

To display or change the timeout value, enter:

Platform	Steps

Linux	`/sbin/sysctl -e net.ipv4.tcp_keepalive_time`
	If necessary, adjust the timeout period.
	For example, to reduce the timeout period to 15 minutes (900 seconds) enter:
	`/sbin/sysctl -w net.ipv4tcp_keepalive_time=900`

34.5 Check for Hardware Errors

Check for hardware errors on a regular basis.

These error messages indicate problems that may lead to database corruption:

- Disk read, write, or retry errors
- Timeouts
- System panics
- Memory problems of any type
- (Linux) Check the `/var/log/messages` file on a regular basis.
  See the operating system documentation for more information.

34.6 Monitor Operating System Resources

Use operating system tools to help monitor performance.

- `iostat` – reports the amount of I/O on terminals and hard disks and how CPU time is spent.
- `vmstat` – monitors virtual memory usage.
- `netstat` – monitors network status.
- `ps` – gives you an accurate snapshot of accumulated CPU time and usage for individual processes. This can be very helpful in determining the dataserver-, engine-, and process-specific loading.
- `time` – can be useful in determining the various user, system, and real-time resources used over a complete run.

For details about these tools, see your operating system documentation.

34.7 Check Database Integrity

Run `dbcc` checks and perform database backups to protect the integrity and recoverability of your SAP ASE databases.

Use `dbcc checkstorage` to run regular consistency checks on large databases when the amount of time you require to perform consistency checks is less than the amount of time required for `dbcc checkalloc` or `dbcc checkdb` to complete. `dbcc checkstorage` performs a fast consistency check, but it is not as thorough as `dbcc checkalloc` or `dbcc checkdb`. See *Preparing to use dbcc checkstorage* and *Checking Database Consistency* in the *System Administration Guide, Volume 2*, and the *Reference Manual: Commands*.

You can also run this sample C shell script, which calls several `isql` scripts:

```
#!/bin/csh -f
 if ( -e dbcc_mail.out) then
   rm dbcc_mail.out
 endif
 foreach i (*.dbcc)
 isql -Usa -Ppassword < $i > dbcc_out
 if ( ‘grep -c ‘Msg 25[0-9][0-9]’ dbcc_out’ ) then
   echo "There are errors in" $i >> dbcc_mail.out
   cat dbcc_out >> dbcc_mail.out
 else
   echo "Backing up " $i:r >> dbcc_mail.out
   isql -Usa -Ppassword < $i:r.backup
 endif
 end
 mail -s "Backup Report" jjones < dbcc_mail.out
```

The first set of scripts (one for each database with a file name appended with `.dbcc`) runs `dbcc checkalloc` and `dbcc checkdb` for each database and sends the messages to an output file called `dbcc_out`.

For example, the script `master.dbcc` runs `dbcc` to check the `master` database:

```
dbcc checkalloc (master)
go
dbcc checkdb (master)
go
```

The C shell script then runs the `grep` command to find 2500-level error messages in `dbcc` output. The results of the `grep` command are sent to an output file called `dbcc_mail.out`.

Next, the script invokes an `isql` backup script for each database for which no 2500-level errors occurred, and adds the “Backing up `<database_name>`” line to `dbcc_mail.out`. For example, the script `master.backup` backs up the `master` database:

```
use master
go
dump database master to master_dump
go
```

You may want to add appropriate `dump transaction` commands to your scripts.

If there are 2500-level error messages, the script does not back up the database. At the end of the script, `dbcc_mail.out` is mailed to the system administrator “jjones”, who then has a record of fatal `dbcc` errors and successful backups.

You can tailor the sample shell and `isql` scripts to suit the needs of your installation.

To have the scripts execute automatically, edit the `crontab` file, and add an entry similar to this, which executes a C shell script called dbcc_ck daily at 2:00 a.m.:

```
00 02 * * * /usr/u/sybase/dbcc_ck 2>&1
```

This example executes a C shell script called `dbcc_ck` at 2:00 a.m. every morning.

35 Set Up Communications Across the Network

Each instance in a cluster can communicate with other SAP ASE servers, SAP Open Server applications, and client software across a network.

Clients can communicate with one or more servers, and servers can communicate with other servers via remote procedure calls.

Directory services contain information about the network locations of servers, as well as entries for all SAP ASE servers, Backup Servers, and other server products on the network.

In the SAP client/server environment, a client can connect with an instance if it knows where the server resides on the network and if the server supports the client's language or character set. When a client initiates a connection, it looks in its directory services for the network location of the target server.

Directory services list the name and address of every server, including Backup Server and XP Server. When you use a client program and you want to connect with a particular server, the client program looks up the server name in the directory services and connects to that server.

Servers also need network information. When a server starts, it looks in its interfaces file to determine where to listen for client connection requests. In addition, each instance can take on a client role when it makes remote procedure calls to other instances.

Related Information

interfaces File Format [page 224]
Create a Master interfaces File [page 227]
Configure the interfaces File for Multiple Networks [page 229]

35.1 How SAP ASE Determines Which Directory Service Entry to Use

SAP ASE uses directory services to determine the address at which to listen for clients.

When you start an instance, it:

1. Looks for the server name supplied in the command line `-s` option. If the server name is not supplied in the command line, it determines its own name by checking the value of the DSLISTEN environment variable. If the DSLISTEN environment variable is not set, SAP ASE assumes that the server name is SYBASE.
2. Looks in directory services for an entry that matches the name found in the steps above.
3. Uses the network information provided by the directory services entry it has found to listen for client connections.

35.2 How a Client Uses Directory Services

Clients use directory services to determine how to connect to a server.

When a client connects to a server, it:

- Determines the name of the server either programmatically or by referring to the DSQUERY environment variable. If the application user has not set DSQUERY, the runtime value for the server name defaults to the SYBASE environment variable.
- Looks in directory services for an entry that matches the name of the server.
- Uses the network information provided by the directory services entry to connect to the server. If the client cannot connect the first time, it makes additional attempts according to the delay and retry numbers indicated in directory services. If no matching entry is found, an error message is written to the client's standard error file. If multiple networks are supported, the client attempts to connect using the information in the second network address entry for the server.

See the *Open/Client Programmer's Supplement* for your client platform or the appropriate Open/Client documentation for a more detailed information about client connections.

35.3 Creating a Directory Services Entry

The `sybcluster` utility creates a directory services entry for the cluster and for each instance when you create the cluster.

You can also use the following utilities to edit the network information in directory services:

- `dsedit` – a GUI utility.
- `dscp` – a UNIX command line utility.

For details on using these utilities, see the *SAP ASE Utility Guide*.

35.4 Supported Directory Drivers

SAP supports several directory drivers.

The supported drivers are:

- Interfaces driver
- LDAP driver
- CDS provided by Distributed Computing Environment (DCE)

See the *Open Client/Server Configuration Guide* for your platform.

35.5 Contents of the interfaces File

An interfaces file contains network information about all servers on your network, including SAP ASE, Backup Server, and XP Server, plus any other server applications, such as Replication Server, and any Open Server applications.

The network information in the file includes the server name, network name, or address of the host machine, and the port, object, or socket number (depending on the network protocol) on which the server listens for queries.

Each entry in an interfaces file can contain two types of lines:

- Master lines – used by server applications to listen for queries over the network. This information is called a *listener service*.
- Query lines – used by client applications to connect to servers over the network. This information is called a *query service*.

The network information contained in the master and query lines for a server is identical because a server listens for connection requests on the same port that clients use to request connections.

A server needs both master and query lines in its `interfaces` file, since servers sometimes act as clients to other servers.

A client's interfaces file does not need a master line. It functions correctly with only a query line.

If Your Site has Multiple Installations

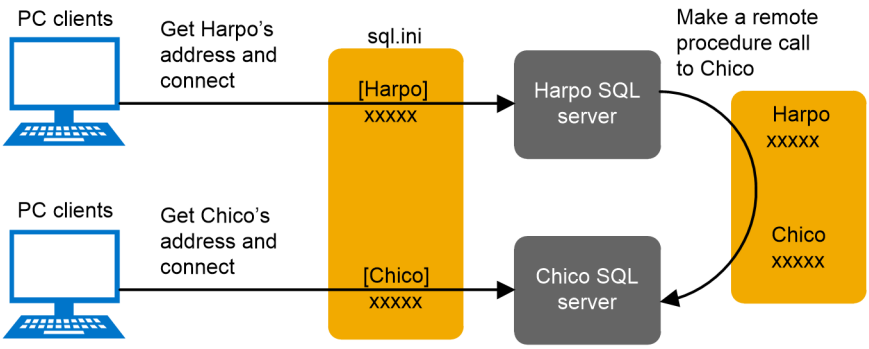
If you have more than one SAP ASE installation, each server's interfaces file should contain information about all servers on the network.

If all of your server products are running on the same platform, you can create one master `interfaces` file and copy that file to each machine.

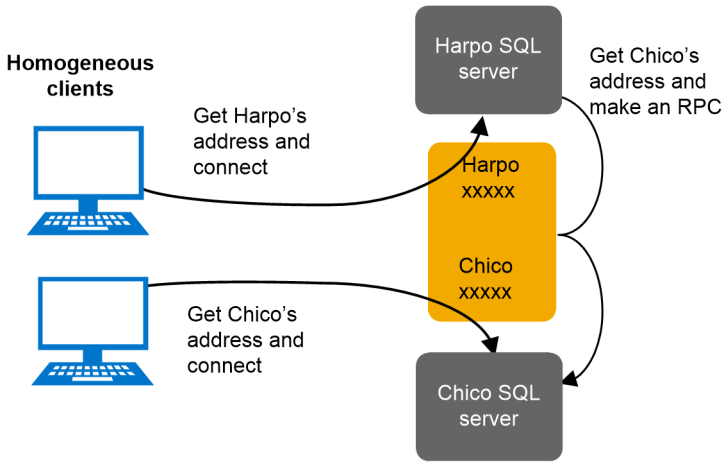
Configure the interfaces files for multiple networks if your host machine supports them.

Heterogeneous and Homogeneous Environments

If the platforms are different, each platform may require a different format and configuration for its `interfaces` file. This figure illustrates how a PC client uses network information in its `sql.ini` interfaces file to connect to an instance running under UNIX, and how an instance uses its `interfaces` file to connect to another server during a remote procedure call.



If both a client and a server are running in UNIX, the same interfaces file is valid for both. This figure illustrates how clients and instances running in a homogeneous environment can use copies of the interfaces file to establish connections. Because the two instances are running under the same operating system, they can use the same interfaces file, or exact copies of the same file.



35.6 interfaces File Format

The format for the `interfaces` file supports the TLI and TCP style entry.

These rules apply:

- Each SAP ASE has only one entry, although there may be multiple lines in the entry.
- Each line that follows the `<servername>` line must begin with a space or a tab.
- Each element on the line must be separated by a single space.
- Each entry is separated by a blank line.
- You can add comments to an interfaces file by adding a pound sign (#) at the beginning of the line and a line break at the end.

There are two `interfaces` file entry formats, TLI and TCP.

> **Note**
>
> SAP ASE does not support the TLI interface in threaded mode.

A TLI-style entry uses this format:

```
<servername> <retry_attempts> <delay_interval><newline>
 <tab><service_type> <api> <protocol> <device> <address> <filter><newline>
 <tab>ha_failover <servername><newline>
```

For example:

```
harpo_server1
```

```
master tli tcp /dev/tcp 0x0002333337f00001
```

```
query tli tcp /dev/tcp 0x0002333337f00001
```

A TCP-style entry uses this format:

```
<servername> <retry_attempts> <delay_interval><newline>
 <tab><service_type> <protocol> <network> <machine> <port filter><newline>
 <tab>ha_failover <servername><newline>
```

For example:

```
chico_server1
```

```
 master tcp ether chico 5678 ssl
```

```
 query tcp ether chico 5678 ssl
```

35.6.1 Components of an interfaces File Entry

The `interfaces` file consists of entries that define server attributes.

Component	Value
`<servername>`	Name of an SAP ASE instance or Backup Server. Requirements for a server name are: • The name cannot be more than 30 characters long. • The first character must be a letter (ASCII a through z, A through Z). • The characters that follow must be letters, numbers, or underscores (_).
`<retry_attempts>`	(Optional) Number of times you want the client to try to connect to a server after initial failure. The default is 0.
`<delay_interval>`	(Optional) Number of seconds between connection attempts. The default is 0.

Component	Value
<service_type>	Type of service defined by entry. Must be one of the following: • master • query
<api>	Application programming interface available to the network. The only supported value is "tli".
<protocol>	The only available protocol type is TCP/IP, represented by the letters "tcp".
<network>	Name of the network; not currently used by SAP ASE. `sybcluster` enters “ether” as a placeholder.
<host>	Network name or address of server’s host machine. For TCP/IP, use either the host name or Internet address. Maximum size of entry is 32 bytes. To determine the host name of a machine, log in to that machine and enter: `/bin/hostname`
<machine>	Network name or address of server’s host machine. You can use either the host name or Internet address. The maximum size of entry is 32 bytes. To determine the host name of a machine, log in to that machine and enter: `/bin/hostname`
<device>	The network device endpoint. For TCP networks, the device varies according to the vendor supplying the networking software. Check the vendor-specific documentation for the name of the device. Your network may provide multiple stream devices corresponding to different protocols in the TCP protocol suite. Choose the TCP streams device. The common TCP streams devices in `/dev/tcp`.
<address>	(For a TLI protocol entry) Address consists of: • Address prefix, “\x” for TLI. • Network type, always 0002. • Port number converted to four digits, hexadecimal. Must be a unique number between 1025 and 65535. Check the `/etc/services` file on each machine on the network to see what port numbers are in use. Enter the SAP ASE port number in a new section of `/etc/services`. You need not make this entry for the operating system to function properly, but the presence of the port number in the file may prevent other users from using the port number. • IP network node address of the host machine converted to 8 digits, hexadecimal. • (Optional) Trailing zeros, 16 digits.

Component	Value
`<port>`	A unique port number between 1025 and 65535. Check the `/etc/services` file on each machine on the network to see what port numbers are in use. Enter the SAP ASE port number in a new section of `/etc/services`. You need not make this entry for the operating system to function properly, but the presence of the port number in the file may prevent other users from using that port number.
`<ha_failover>`	An entry created in directory services or the interfaces file for high availability.
`<filter>`	SAP ASE supports Secure Sockets Layer (SSL) as a filter that is appended to the master and query lines of the directory services. SSL is the standard for securing the transmission of sensitive information.

Related Information

Adding SSL Common Name [page 210]

35.7 Create a Master interfaces File

A master `interfaces` file contains entries for all SAP servers on the network.

The master `interfaces` file can be used with every server and client connected to the network. By distributing copies of a master `interfaces` file, you can ensure that all SAP products on the network interact with one another. Distributing copies of one interfaces file (a master file) with entries for all servers is the easiest way to maintain consistency in the interfaces files in a homogeneous environment on a network.

You can make all changes to one version of the file and then copy the updated master file to all appropriate `Sybase` directories.

Make a master file using either:

- `dsedit` or `dscp`, or,
- A text editor

35.7.1 Using dsedit or dscp to Create a Master interfaces File

Use the `dsedit` or `dscp` utility to create a master `interfaces` file, which you can then distribute to all servers.

Context

If you are not an experienced SAP user, `dsedit` or `dscp` is easier to use than a text editor. Using `dsedit` or `dscp` also ensures that your `interfaces` file is consistent in format.

Procedure

1. Select the `interfaces` file that contains the most complete, up-to-date information.
2. Begin a `dsedit` or `dscp` session in your most recently updated $SYBASE installation.
3. Add entries for any SAP ASE servers or Backup Servers that are not listed in this file.

   See the *SAP ASE Utility Guide*.

35.7.2 Using a Text Editor to Create a Master interfaces File

Use an ASCII text editor to modify a concatenated `interfaces` file.

Procedure

1. Concatenate all individual `interfaces` files.
2. Make a copy of the file.
3. Use an ASCII text editor to modify the copy of the concatenated file.

   When you manually edit an `interfaces` file, be sure that, for each entry, each line following the first line begins with a <tab> character.

   These elements must be correct and unique in the resulting file:
   - `<servername>` – each server entry in the `interfaces` file must be unique. During the `srvbuild` session, you had the choice of entering a server name or accepting the default server name, SYBASE, for those servers. Find any duplicate SYBASE entries in your merged file, and rename them.
   - A combination of the host machine's network name or address and the SAP ASE port or object number.
   - If the original `interfaces` file was created when there was only one machine on the network, its entries may have the word "loghost" in place of the machine name (address). Replace `<loghost>` with the machine name.

35.8 Configure the interfaces File for Multiple Networks

On some platforms, SAP ASE can accommodate multiple networks. This allows SAP ASE to listen for clients over multiple network interfaces. You must add an entry for each network interface to the `interfaces` file.

35.8.1 Configuring the Server for Multiple Network Handlers

Configure multiple network listeners.

Procedure

1. Define a unique host name for each network interface in your operating system's host database.
2. In your `interfaces` file, use a text editor to add copies of the "master" line for your SAP ASE; one for each additional interface you want the server to listen on.
3. Include a unique host name on each line to configure a network handler for each network interface.
4. Port numbers within the interface need not be the same, but they can be. They fall under the same rules for naming and numeric range as the primary network interface.
   This example shows an `interfaces` file for an SAP ASE with two network interfaces. The server host machine is known as SERV_CORPNET on the corporate network and SERV_ENGNET on the engineering network.

   ```
   # PRODUCTION server with two network listeners
    PRODUCTION<tab>3<tab>3<newline>
    <tab>master tcp ether SERV_CORPNET 4559
    <tab>master tcp ether SERV_ENGNET 5479
    <tab>query tcp ether SERV_CORPNET 4559
   ```

   When SAP ASE restarts, it spawns a network handler process for each master line in the entry that corresponds to the server's DSLISTEN value. Connections made on each interface are handled equally, as peers.

35.8.2 Configure Client Connections

When an SAP ASE client scans the interfaces file for a server name, the client uses the first "query" entry it encounters for the server's entry. This makes configuring clients to use multiple network connections less straightforward than configuring the server ports.

You have two options:

- Use the same DSQUERY name for all clients. The interfaces files on the different machines contain different network names.
- Use different DSQUERY names for the clients. The interfaces files on all the machines are the same, but they contain multiple DSQUERY names.

35.8.2.1 Manage One Network-Independent DSQUERY Name

You can maintain one DSQUERY name for all clients, on all networks, and alter each network's interfaces file accordingly.

If uniform client DSQUERY naming is important, you can make the necessary changes in the network addressing of the clients in the interfaces file. You can install separate $SYBASE installation directories and distinct interfaces files on client file servers on each network to allow users to connect to the correct network address. Instead of altering the DSQUERY name the clients use, you maintain one DSQUERY name for all clients, on all networks, and alter each network's interfaces file accordingly.

This method assumes that:

- You have complete control over what the SAP installation clients see on each network.
- The interfaces file (at the very least) is not shared or copied among SAP installations on different networks.

The interfaces file looks like this on the "engineering" network:

```
PRODUCTION<tab>3<tab>3<newline>
<tab>query tcp ether SERV_ENGNET 5470
<tab>master tcp ether SERV_CORPNET 4559
<tab>master tcp ether SERV_ENGNET 5479
```

The interfaces file looks like this example on the "corporate" network:

```
PRODUCTION<tab>3<tab>3<newline>
<tab>query tcp ether SERV_CORPNET 4559
<tab>master tcp ether SERV_CORPNET 4559
<tab>master tcp ether SERV_ENGNET 5479
```

The "query" line in each file name is different, depending on the network to be used.

The full "master" entry is present in both files. This is allowed because only SAP ASE uses the "master" lines. Assuming that the server host machine can see both networks (both host names are interchangeable), it does not matter which interfaces file is used for SAP ASE start-up.

35.8.2.2 Using Different DSQUERY Names

You can use different DSQUERY names for each network listener.

Procedure

1. Choose an additional server name.

   You can concatenate the original server name and the network name. For example, if your server is named PRODUCTION, you could choose the names PRODUCTION_network1 and PRODUCTION_network2.
2. Do one of the following:
   - (PC clients) Use `sqledit` to create multiple `sql.ini` file entries for the server, one for each network. In the following example, you create one entry for PRODUCTION_network 1 and one for

PRODUCTION_network2. For more information, see the Open Client documentation for your client platform.

- (UNIX clients) Edit the `interfaces` files with the ASCII editor. From the server's `interfaces` files, copy the server name line and the "master" line for each network into the client interfaces file. Add the appropriate server name for each entry, and change "master" to "query."

Clients on each network use the DSQUERY value that corresponds to the network the client is on. The following example can use either PRODUCTION_network1 or PRODUCTION_network2:

```
# Client entry for PRODUCTION on network1
PRODUCTION_network1<tab>3<tab>3<newline>
<tab>query tcp ether serv_corpnet 4559
# Client entry for PRODUCTION on network2
PRODUCTION_network2<tab>3<tab>3<newline>
<tab>query tcp ether serv_engnet 5479
```

35.8.3 Configuring for Query Port Backup

If a client is connected to a server via two networks, the client can establish a connection via the second network if the first one goes down.

Procedure

1. Install multiple "master" and "query" lines in a server entry in the `interfaces` file.
2. SAP ASE listens for connections at both ports. Clients looking for a host name and a port number for a connection to an SAP ASE try the port on each "query" line, in order, until they establish a connection.

   This example shows how to configure a backup network for use only when the normal connection fails. The primary network is "corporate network" and the backup network is "engineering network".

   ```
   # PRODUCTION server with two network listeners
   PRODUCTION<tab>3<tab>3<newline>
   <tab>master tcp ether SERV_CORPNET 4559
   <tab>master tcp ether SERV_ENGNET 5479
   <tab>query tcp ether SERV_CORPNET 4559
   <tab>query tcp ether SERV_ENGNET 5479
   ```
3. Configure PC-client `interfaces` files with the appropriate multiple "query" entries, as described in the Open Client documentation. For client `interfaces` files in a homogeneous environment, you can copy the entire file entry for the SAP ASE into the client `interfaces` file.
4. A connection on the secondary port occurs only if the corporate network is disabled, or if the corporate network interface on the host machine fails or is shut down due to a network-related failure.

35.9 IPv6 Technology

SAP ASE supports IPv6 technology.

IPv6 terminology:

- Link-local address – an IPv6 address that is usable only over a single link.
- Site-local address – an IPv6 address that can be used within a single site.
- Global address – an IPv6 address that can be used across the Internet.

Interfaces files also provide IPv6 support. An example `interfaces` file entry is:

```
RHAM0_10778_XP  query tcp sun-ether fd77:55d:459d9:169:250:56ff:feb3:4246 19560
 master tcp sun-ether fd77:55d:59d9:169:250:56ff:feb3:4246 19560
```

IPv6 application types:

- IPv6-unaware – an application that cannot handle IPv6 addresses.
- IPv6-aware – an application that can communicate with nodes that do not have IPv4 addresses. In some cases, this might be transparent to the application, for instance when the API hides the content and format of the actual addresses.
- IPv6-enabled – an application that, in addition to being IPv6-aware, takes advantage of some IPv6 features.
- IPv6-required – an application that requires some IPv6 features and cannot operate over IPv4.

35.9.1 IPv6 Infrastructure

A dual-stack infrastructure implements both IPv4 and IPv6. This is the SAP-recommended infrastructure implementation for using SAP ASE as an IPv6-aware server.

SAP applications are IPv6-aware.

Table 11: IPv6 Support

Platform	SAP ASE IPv6 Awareness	Open Client/Server IPv6 Awareness
Linux RHEL 3.0	15.0	15.7

Many SAP products that are based on Open Client/Server, such as XP Server, Backup Server, Replication Server, and Open Switch, became automatically IPv6-aware due to the layered Open Client Transport Control Layer (DB-Library-based), which is IPv6-aware for network-socket operations.

> **Note**
>
> DBlib-based Open Client products are not IPv6-aware.

For SAP ASE, being IPv6-aware is a complex issue because some third-party components within SAP ASE are not IPv6-aware. The functional mechanisms of SAP ASE that are IPv6-aware with respect to the supported platforms and versions include:

- Connection handler

- RPC mechanisms
- Job Scheduler task/agent session connection
- Network host API
- UDP message support for `sybsendmsg`
- Component Integration Services connectivity
- Host/name resolving
- XML URL connection handler
- Auditing for client address data

These functional mechanisms in SAP ASE do not currently support IPv6, but are scheduled to do so in future versions:

- Java support
- License Management Server
- LDAP driver
- Private interconnect between instances in a cluster

By default, SAP ASE is IPv6-unaware.

You must start the SAP ASE Cluster Edition with trace flag 7841 to make it IPv6-aware. This causes the SAP ASE Cluster Edition to determine IPv6 availability and makes the SAP ASE Cluster Edition IPv6-aware.

See your network or IT specialist to configure your platforms and network infrastructure correctly for IPv6 support.

To capture and log address connection requests and host/name lookups, set a second trace flag, 7815, when you start the SAP ASE Cluster Edition.

The IPv6-related SAP ASE Cluster Edition trace flags are:

- T7841 – enables the SAP ASE Cluster Edition IPv6-awareness.
- T7815 – reports all the SAP ASE Cluster Edition IPv4 and IPv6 Client address connect requests.

Before starting SAP ASE for IPv6-aware operations, make sure your infrastructure is configured correctly. After your operating system is set up correctly, you can configure and enable an IPv6 `connection handler` by adding an additional DCL entry. A single SAP ASE configuration can typically carry up to 32 `connection handler` assignments within the DCL.

For example, if you have a site-local setup with two domains administered under the names:

```
sybase.com - being responsible for all IPv4 networking applications
v6.sybase.com - being responsible for all IPv6 networking applications
```

The DCL entry named "SYBASE" on the host "revival" for port 17100 looks similar to:

```
SYBASE
master tcp ether revival.sybase.com 17100
query tcp ether revival.sybase.com 17100
master tcp ether revival.v6.sybase.com 17100
query tcp ether revival.v6.sybase.com 17100
```

In this example, when SAP ASE is started with IPv6-awareness, it creates two `connection handlers`. One listens on port 17100 for incoming IPv4 client connection requests, and the other listens on port 17100 for incoming IPv6 client connection requests.

> **Note**
>
> When you start SAP ASE, set trace flag 7815 to capture and log IPv4 and IPv6 client address connection requests and host and name lookups.

36 Troubleshoot

You can troubleshoot common errors.

36.1 Verifying the Cluster Environment

Many errors that occur using the SAP ASE Cluster Edition result from configuration problems in the cluster environment.

Before you configure your cluster:

- Verify that you have set your environment variables by sourcing the `SYBASE.csh` or `SYBASE.sh` file located in `$SYBASE`.
- Verify that all required libraries are installed on the host by running `dataserver -v` from each node.
  If any system libraries are missing, you see an operating system error, and the dataserver version number does not appear. Correct this problem before proceeding. If `dataserver` displays the version string without error, you can assume that all required system libraries are installed.
- Verify that each node in the cluster can read from and write to each database device by using the operating system `ls -l` command, then testing the readability of a device:

  ```
  dd if=<device path> of=/dev/null count=x
  ```

  You should see a result similar to:

  ```
  %dd if=/dev/raw/raw123 of=/dev/null
  count=10
  10+0 records in
  10+0 records out
  ```

  While you can also use the `dd` utility to test the writeability of your devices as well, do this only if there is no data in the devices to preserve.
- Run the `ping` utility to verify connectivity among all nodes. From each node, attempt to ping the host name or network address of every other node. Do this for each network to be used. For example, if your configuration uses a public and two private networks, verify that `ping` succeeds for all combinations of node and network address.
  Determine the private interconnect addresses used by each instance. Use the `sybcluster 'show cluster config'` parameter. For example, if your cluster contains nodes node1 and node2, `sybcluster` displays information similar to this:

  ```
  SYBCE> show cluster config
  ** Cluster configuration for "SYBCE" **
      Interface Path "/sybce"
      Trace Flags:
      There were are no trace flags.
      Maximum Instances "4"
      Quorum "/dev/raw/raw23"
      Master Device "/dev/raw/ra24"
        logfile INSTANCE1 /sybce/ASE-16_0/install/GATEST_INSTANCE1.log
  ```

```
      run_parameters INSTANCE1
      logfile INSTANCE2 /sybce/ASE-16_0/install/GATEST_INSTANCE2.log
      run_parameters INSTANCE2
    Primary Interconnect "udp"
      Server[1] INSTANCE1 node1_priv 49152 49171
      Server[2] INSTANCE2 node2_priv 49172 4919
```

`node1_priv` and This cluster includes interconnect network addresses `node2_priv`. From node1, execute `ping node2_priv` to verify that the address of the private network on node2 is accessible from node1. From node2, execute `ping node1_priv` to verify that the private network on node1 can be reached from node2.

If the `ping` command fails or error messages indicate a problem with the private network, check that:

- The information contained in the `/etc/hosts` file.
- ThisThe condition of the network cables, routers, or switches used by the private networks.
- The names or IP addresses specified in the cluster configuration reported by the `sybcluster` "`show cluster config`" command.

36.2 Restarting the Cluster Using a dataserver Binary from an Earlier Version of SAP ASE

The SAP ASE Cluster Edition does not start and writes an error message to the error log.

The message is: "`Cluster is running with message version <y>. This version of ASE requires message version <x>`".

This may happen when:

- You attempt to use different versions of the `dataserver` binary in the same running cluster.
- You apply an EBF or ESD without first gracefully shutting down a cluster using an earlier version of the `dataserver` binary.

Troubleshoot using the following steps:

- Start an instance of the cluster using the old version of the `dataserver` binary, then issue `shutdown cluster`.
- Restart the cluster using the new version of the `dataserver` binary.
- If the previous version of the `dataserver` binary is unavailable, resolve this error message by re-creating the quorum device:

  > **⚠ Caution**
  >
  > These steps bypass SAP ASE safety checks. Make sure that there are no instances running while you perform them.

  1. Verify that all instances of the cluster are shut down.
  2. Extract the cluster input file information:

     ```
     $SYBASE/ASE-16_0/bin/qrmutil --quorum_dev=<path_to_quorum>
     --extract_config=quorum.out
     ```

3. Start the SAP ASE Cluster Edition using the new `dataserver` binary, and rebuild the quorum device:

```
dataserver --quorum_dev=<path_to_quorum> --instance=<instance_name>
--buildquorum=force --cluster_input=quorum.out
```

4. Shut down the cluster.
5. Start the cluster using normal procedures.

> **Caution**
>
> Use the `--buildquorum` or `--cluster_input` `dataserver` parameters only for these steps. Do not use them for subsequent cluster or instance restarts.

36.3 Errors Accessing Disk Devices

Configure all disk devices used by the SAP ASE Cluster Edition so that they are accessible from all the nodes in the cluster. The paths to these devices must be the same on all nodes, and the account used to start the cluster must have permission to read and write to all of the disk devices.

If the SAP ASE Cluster Edition reports that any of the devices cannot be accessed, verify from each node in the cluster that:

- The device paths specified when configuring the cluster are accessible from all nodes in the cluster.
- The account used to start the cluster has permission to read and write to these devices.
- The device paths are the same on all nodes in the cluster.
- Any symbolic links used to refer to the devices are correct.

Verify paths and file permissions with the UNIX `ls` and `ls -l` commands. Use the UNIX `dd` utility to verify that the devices can be read and written to by the Sybase account.

Related Information

Verifying the Cluster Environment [page 235]

36.4 Verifying That the Cluster Is Down

The `sybcluster` utility may not be able to determine whether the cluster is running if the cluster has crashed and left some status information in the quorum device in an inconsistent state.

Context

If you are uncertain about the status of the cluster, analyze each instance in the cluster to determine the cluster's status.

Procedure

1. Use `isql` to log in to each instance in the cluster.
2. Use `sybcluster 'show cluster status'` to view the status of each instance in the cluster.
3. The instance may be down if `sybcluster` does not report that the state of all instances is `'Up'` and that the `heartbeat` is `'Yes'`.
4. Use the UNIX `ps` command on each node in the cluster to determine whether processes representing the `dataserver` program are running for each instance configured for that node.
5. After starting the first instance, use `sybcluster start cluster` to start all remaining instances in the cluster.

Next Steps

If you determine the instances are not running, unlock and restart the cluster:

```
start instance <instance_name> unlock
```

36.5 Creating Cluster Using sybcluster Fails With Error -131

The `sybcluster create cluster` command may fail with error -131 and issue an error message.

The message, which states that the parent directory in which the raw devices are defined cannot be accessed, looks similar to:

```
INFO - Choosing the first instance to be created using the connected agent...
ERROR - Parent directory access error. The parent directory /dev/rdsk for the
device can not be accessed.  Please change the protection on the device and try
again.
```

```
INFO - Create cluster error: -131
```

You may have incorrectly spelled the name of the raw device. Check the complete name of each raw device and verify that it is correct.

36.6 Cluster Creation Fails Leaving Files in $SYBASE Directory

If the `sybcluster create cluster` command or SAP Control Center terminates with an error condition, some files may be left in the cluster installation directory, and you may need to remove them before attempting to create the cluster again.

Context

Only steps 1 and 2 are required before you can initiate another cluster creation operation. However, perform the other steps to reduce the number of files in the installation area.

Procedure

1. Remove any entries for the cluster in the interfaces file in the installation directory.
2. Remove SCC plug-in directories for each node from these locations:

   ```
   $SYBASE/SCC-3_3/nodes/<node_name>/plugins/<cluster_name>
   ```

3. Remove resource files with names ending in `.res` from the `$SYBASE` directory.
4. Remove cluster definition files with names ending in `.inp` from the `$SYBASE` directory.
5. Remove the intermediate `RUN_instance_name` file from `$SYBASE/ASE-<version_number>/install`.
6. Remove any error log files from `$SYBASE/ASE-<version_number>/install`.
7. Remove any instance configuration files from `$SYBASE`.

36.7 SCC Agent Starts but sybcluster connect Fails

The `sybcluster 'connect'` command may fail during the SCC agent start-up or when executing the `sybcluster` command because of incorrect network configuration information.

- Examine the SCC agent log files for each agent in the cluster to determine whether any errors are reported when opening the RMI listener for the agent. The agent log files are located in: `$SYBASE/SCC-<version_number>/nodes/node_name/log/agent.log`

- Specify the correct node names and listening port numbers for the `-F` parameter when starting `sybcluster`:

```
sybcluster -U uafadmin -P -C MYCLUSTER
-Fnode1:1234,node2:1234
```

36.8 Disk Devices in Use on Linux

By default, some devices are bound to a disk. If you attempt to create a cluster and inadvertently specify one of these devices, `disk init` fails and the Create Cluster wizard or `sybcluster` cannot create the cluster.

Before you start the Create Cluster wizard or a `sybcluster` session, verify that the disk device is available.

Related Information

Verifying the Cluster Environment [page 235]

36.9 Instances Fail to Join the Cluster

Instances may fail to join the cluster because of problems with the private networks used to support interconnect communication.

If one or more instances in the cluster fail to start:

- Examine the error logs for those instances that could not join to see whether any messages indicate a network communication failure. The error logs are located in `$SYBASE/$SYBASE_ASE/install/instance_log`.
- Use the `ping` utility to verify that all private networks used on the nodes used by the cluster are accessible from all nodes in the cluster.
- Check for errors in the SCC agent logs.

Related Information

Verifying the Cluster Environment [page 235]

36.10 Verifying RDMA Support

Steps used to ensure that RDMA is present on the system.

Prerequisites

Install the following packages:

- libibverbs
- librdmacm

Procedure

1. Check the presence of InfininBand devices using commands `ibv_devinfo` and `ibv_devices`.

   `ibv_devices` provides concise output and gives information about all RDMA devices. `ibv_devinfo` expands on the basic information about all RDMA devices.

   For example:

   ```
   $ ibv_devices
       device                 node GUID
       ------              ----------------
       mlx4_0              f04da29097791164
   ```

2. From `ifconfig`, verify that there is an entry for the InfiniBand interface. The interface appears as an `ib*`, where * can be any number, dependent of the number of host channel adapters installed.

   ```
   $ ifconfig
   eth0      Link encap:Ethernet  HWaddr A4:BA:DB:4F:3B:14
   ...
   ib0       Link encap:InfiniBand  HWaddr 80:00:00:48:FE:
   80:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
             inet addr:10.0.1.22  Bcast:10.0.1.255  Mask:255.255.255.0
             inet6 addr: fe80::f24d:a290:9779:1165/64 Scope:Link
             UP BROADCAST RUNNING MULTICAST  MTU:2044  Metric:1
             RX packets:150686 errors:0 dropped:0 overruns:0 frame:0
             TX packets:10710 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:256
             RX bytes:20345868 (19.4 MiB)  TX bytes:1517476 (1.4 MiB)
   ```

   View the associated name to the `inet addr` from `/etc/hosts`.

   ```
   $ cat /etc/hosts
   127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
   ::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
   ...
   # infiniband network
   10.1.1.1        star-ib-x
   10.2.1.1        star-ib-y
   10.0.1.1        star-ib
   10.0.1.2        star1-ib
   ```

```
...
10.0.1.21       asedg1lnx1-ib dg1-ib
10.0.1.22       asedg1lnx2-ib dg2-ib
10.0.1.23       asedg1lnx3-ib dg3-ib
10.0.1.24       asedg1lnx4-ib dg4-ib
...
```

3. Use the rping utility to verify that the RDMA protocol is fully functioning through the above address.

```
$ rping -s -a asedg1lnx2-ib -v &
[1] 22603
$
$ rping -c -C 5 -a asedg1lnx2-ib -v
server ping data: rdma-ping-0:
ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqr
ping data: rdma-ping-0: ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqr
server ping data: rdma-ping-1:
BCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrs
ping data: rdma-ping-1: BCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrs
server ping data: rdma-ping-2:
CDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrst
ping data: rdma-ping-2: CDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrst
server ping data: rdma-ping-3:
DEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstu
ping data: rdma-ping-3: DEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstu
server ping data: rdma-ping-4:
EFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuv
ping data: rdma-ping-4: EFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuv
server DISCONNECT EVENT...
wait for RDMA_READ_ADV state 9
client DISCONNECT EVENT...
[1]+  Done                    rping -s -a asedg1lnx2-ib -v
$
```

36.11 SAP ASE Boot Errors with RDMA

SAP ASE boot-up fails with errors under certain conditions.

This happens if the `ulimit` for `max lock memory` is not high enough. The `ulimit` must high since RDMA locks the entire memory registered with it, otherwise, you may encounter these errors:

```
02:0000:00000:00000:2015/10/09 01:25:54.01 kernel Attempt to create RDMA
completion queue failed in ibv_create_cq errno : 12
02:0000:00000:00000:2015/10/09 01:25:54.01 kernel Failed to initialize RDMA
subsystem. This is a fatal error.
02:0000:00000:00000:2015/10/09 01:25:54.01 kernel Cannot initialize the Primary
cluster interconnect.
02:0000:00000:00000:2015/10/09 01:25:54.01 kernel The cluster interconnect
protocol has not been defined in the cluster input file.
02:0000:00000:00000:2015/10/09 01:25:54.01 kernel cipc_init failed.
```

If this occurs, check the `max lock memory` using `ulimit -l`, and if it is not enough, increase it or set it to `unlimited` by editing /etc/security/limits.conf:

```
$ ulimit -l
64
Adding below lines in /etc/security/limits.conf will make it unlimited.
@sybase         hard    memlock        unlimited
@sybase         soft    memlock         unlimited
```

```
$ ulimit -l
Unlimited
```

If the secondary interconnect is configured, an error also occurs. Configure the cluster to not use secondary interconnect if you encounter this error:

```
01:0000:00000:00000:2015/11/08 22:19:41.74 kernel  Secondary network is not
supported when using the RDMA protocol.
01:0000:00000:00000:2015/11/08 22:19:41.74 kernel  Failed to load the cluster
configuration.
01:0000:00000:00000:2015/11/08 22:19:41.74 kernel  cluster_early_init failed
```

36.12 Private Interconnect Failure

The cluster does not start if a private interconnect is not configured correctly. Use the operating system `ping` command to verify the private interconnect is working. If `ping` does not work, see your system administrator to enable the interconnection communication between various nodes.

Related Information

Verifying the Cluster Environment [page 235]

36.13 Client Connection Failover Fails

This error occurs when a client connection failover fails when the client connects to the cluster using IP addresses and the nodes are not included in the local `/etc/hosts` or DNS.

SAP ASE sends the failover instance addresses as they are listed in the cluster's interfaces file. If the cluster's interfaces file lists the instance network addresses as host names, SAP ASE returns the host names to the client applications. However, the client application uses DNS or the `/etc/hosts` file to resolve the names of the hosts for the cluster instances, so failover is unsuccessful if the clients do not have the host names in their `/etc/hosts` file or DNS server.

- Verify that all nodes in the cluster are listed in the client system's DNS server or the client system's `/etc/hosts` file.
- Use IP addresses rather than host names in the cluster's interfaces file.

36.14 Client Fails to Reconnect to Alternate High-Availability Servers

Clients with DNS name resolution problems may fail to reconnect to an alternate high-availability server when the instance to which it is connected fails.

The message is similar to:

```
Connection to Sybase server has been lost, connection to the next available HA
server also failed. All active transactions have been rolled back.
```

DNS name resolution issues occur when the client's host cannot correctly resolve a `<hostname>` for the alternate instances.

To determine if this error is produced by a DNS name resolution problem, issue a `ping <hostname>` command for each alternate instance in the high availability list, where `<hostname>` is the host on which the alternate servers reside. If the `<hostname>` cannot be resolved, or resolves to the wrong IP address, rectify the name resolution issue according to your system administration documentation.

36.15 sybcluster Cannot Connect If All Connections Use SSL

If all SAP ASE listening ports use SSL, SAP ASE issues an error message.

The error message is similar to:

```
2008-03-20 10:42:46,260 ERROR [Timer-6] GA1:GA1_1:SQLConnect:270:Login Failure -
The user "sa" and the entered password is not authorized to connect to the
cluster.
```

Verify that there is a non-SSL connection for each instance in the cluster and that this connection is included in the interfaces file that SCC agent uses.

36.16 jConnect Sample Disables HA

The sample jConnect connection string for the SAP ASE Cluster Edition is incorrect.

The incorrect string is:

```
URL="jdbc:sybase:Tds:server1:port1,server2:port2,...,serverN:portN/mydb?
JCONNECT_VERSION=6&PACKETSIZE=1024&DYNAMIC_PREPARE=true&REQUEST_HA_SESSION=true"
```

This string fails because the `JCONNECT_VERSION=6` parameter emulates the high availability companion server functionality. The JDBC driver causes a client Java exception when the client tries to use the jConnect driver's Failover property but does not have a high availability failover server specified in the connection string.

Use this correct connection string:

```
URL="jdbc:sybase:Tds:server1:port1,server2:port2,...,serverN:portN/mydb?
&PACKETSIZE=1024&DYNAMIC_PREPARE=true&REQUEST_HA_SESSION=true"
```

36.17 PC-Client Installation – java.lang.NoClassDefFound Error

This error occurs on the Windows platform when you unpack the PC Client tar file using the MKS `tar` utility rather than the `Winzip` utility. On Windows platforms, the MKS `tar` utility truncates full path names, resulting in missing files.

This issue occurs on UNIX platforms if you do not use the GNU tar utility to untar the SAP ASE installer.

For Windows platforms, always use the `Winzip` utility to unpack the installer. For UNIX platforms, always use GNU tar utility to untar the installer.

36.18 The Cluster Entry "name" Did Not Contain Any Servers

If the quorum device is not accessible to the SCC agents, errors may occur when you issue the `sybcluster start cluster` and `start instance` commands.

The errors are:

```
start cluster
ERROR - The cluster entry SDCDEMO did not contain any servers
```

```
start instance INSTANCE1
ERROR - The cluster entry SDCDEMO did not contain any servers
```

- The SCC agent may not have permission to read the quorum device. Verify that you started the SCC agent under the proper user account and that this account has permission to both read and write the quorum device.
- `sybcluster` and the SCC agent are using an incorrect path to the quorum device on one or all nodes in the cluster. If you used symbolic links, verify that all links are correct on all nodes.
- Another process is preventing the SCC agent from reading the quorum device. This could be caused by functional problems or configuration errors in the disk storage system.

36.19 After Password Change, sybcluster Cannot Manage the Cluster

`sybcluster` uses the SCC agent to connect to and perform operations on the cluster (for example, `shutdown cluster`). The SCC agent logs in to the cluster, and requires the correct sa login and password to do this.

The sa login and password are stored in encrypted form in the SCC plug-in `configure` file for the cluster. Use `sybcluster set cluster login` to change the login and password that SCC uses to connect to the cluster.

Any time you change the SAP ASE password, you must also change the SCC agent login.

Perform one of the following to change the login and password that SCC uses to connect to the cluster:

- Use `sybcluster set cluster login` to set the SCC login, password, or both. Connect to the cluster before issuing this command. The syntax is:

  ```
  set cluster login sa-login [password sa-password
  ```

- Use SAP Control Center to modify the SCC plug-in.
- Shut down all cluster SCC agents, then edit the SCC configuration files.
  The SCC SAP ASE login and password are stored in:

  ```
  $SYBASE/SCC-3_3/nodes/<node name>/plugins/
  <cluster name>/agent-plugin.xml
  ```

  For example:

  ```
  <set-property property="ase.user" value="sa" /><set-property
  property="ase.password"
  value="REVTe1NZVUFGfWNvbS5zdW4uY3J5cHRvLnByb3ZpZGVyLlN1bkpDRXtTWVVBRn1jUEZPSkJ
  oTTZ2QT0=" />
  ```

  > **Note**
  >
  > If the password tag is missing, use SAP Control Center (see previous step).

  Enter the password in plain text. `passencrypt` generates an encrypted text string. Enter this entire string between the quotation marks in the `password` tag in the `agent-plugin.xml` file.
  Generate the encrypted password:

  ```
  $SYBASE/SCC-3_3/bin/passencrypt
  ```

36.20 Agent "cannot be found"

`sybcluster show agents` does not display all agents.

Often, host systems names in a cluster are similar.

If you use `sybcluster -F` to specify the connections to each of the agents in your cluster, verify that each host system is spelled correctly, that you have specified port numbers, and that the port numbers are correct.

36.21 SCC Plug-In Register Error

You may see an SCC plug-in register error because of an existing cluster parameter value that is inconsistent with a newer configuration parameter.

The error message looks similar to:

```
2008-06-16 14:05:16,051 ERROR [main] Failed to register plugin
com.sybase.ase.cluster_15.0.1. Class
com.sybase.ua.plugins.ase.cluster.ASEClusterAgentPlugin not found. Ignored.
java.lang.ClassNotFoundException:
com.sybase.ua.plugins.ase.cluster.ASEClusterAgentPlugin
2008-06-16 14:05:16,052 INFO [main] Finished loading primordial services.
2008-06-16 14:05:16,063 WARN [main] Bootstrap completed with 1 error(s):
2008-06-16 14:05:16,064 WARN [main] Failed to register plugin
com.sybase.ase.cluster_15.0.1. Class
com.sybase.ua.plugins.ase.cluster.ASEClusterAgentPlugin not found. Ignored.
java.lang.ClassNotFoundException:
com.sybase.ua.plugins.ase.cluster.ASEClusterAgentPlugin
```

If you get this error, it is likely that:

- The SCC agent will not start, and the `agent.log` file displays `Bootstrap completed with x error(s):` near the end of the file.
- You have created, dropped, and reconfigured the same cluster a number of times.

This error message appears in the `agent.log` file or in the terminal window if you did not execute the `scc.sh` as a background process. Perform one of the following tasks:

- Clean up the older parameters that may be introducing the error.
- Remove the `cluster` directory in `$SYBASE_UA/nodes/<node_hostname>/plugins`. You can permanently delete the folder or relocate it outside of the directory structure.

> **Note**
>
> Do not remove the `*snmp` and `*sysam` folders.

After you have performed either of these steps, restart the SCC agent. You may need to manually redeploy the plug-in.

> **Note**
>
> If these steps removed the cluster's `plugin` directory from one of the nodes, use `deploy plugin` to re-create this directory. The cluster requires that each node of the cluster include a `plugin` directory.

36.22 Data On Disk Unavailable: Problems Affecting Database Creation

Disk labels are stored in block 0 of every disk. If raw data slices used for device creation start on block 0, they may overwrite the disk label and make data on the disk inaccessible.

Do not create raw disk slices that start on block 0.

36.23 sybcluster Cannot Find interfaces File

`sybcluster` may report an error message, stating that it cannot find the interfaces file in a nonexistent directory.

```
ERROR - The interfaces file /remote/ase_cluster/sdclinux/SCC-3_3/bin/interfaces
could not be found.
```

This error generally occurs when one or more SCC agents cannot access the quorum device. This may occur because:

- The quorum device does not exist at the configured location.
- The quorum device's permissions do not allow the SCC agent to read or write data.

To resolve this issue, verify that:

1. You sourced the `SYBASE.sh` or `SYBASE.csh` files before starting the SCC agent on each node in the cluster.
2. The quorum device exists and that the login that was used to start the SCC agent has read and write permissions on it.

36.24 When the Server Fails to Start

If the server fails to start, check that the port number specified in the interfaces file is available.

Context

When a server fails to start, and issues this message, the port number specified in the interfaces file may be in use:

```
00:00000:00002:2003/09/22 12:37:23.63 kernel  network name SERV_CORPNET, type
ether, port 4559, filter NONE
00:00000:00002:2003/09/22 12:37:23.65 kernel  ninit: bind, Address already in use
00:00000:00002:2003/09/22 12:37:23.68 server  Error: 1602, Severity: 18, State: 2
00:00000:00002:2003/09/22 12:37:23.68 server  Unable to initialize network 0
```

```
00:00000:00002:2003/09/22 12:37:23.68 kernel  ninit: All master network
listeners have failed.  Shutting down.
00:00000:00002:2003/09/22 12:37:23.68 kernel  ueshutdown: exiting
00:00000:00016:2003/09/22 16:11:35.46 server  SQL Server shutdown by request.
```

Procedure

1. Investigate the port assignment:
   a. Look in the interfaces file to identify the port number assigned to the server.
   b. Determine whether another process is using the same port number by entering:

      ```
      netstat -a
      ```

      If the port number is presented as a local address in the `netstat` output, you cannot use that port for the server. Another process is already using that port.
   c. To verify that the server port is in use, start the server manually. The server does not start if its assigned port number is already in use.

   See the installation documentation for your platform and the *Utility Guide*.
2. If a stale server process is retaining use of the port number, either:
   - Use the operating system `kill` command to terminate the process, or,
   - Use another port number for the server by modifying the interfaces file.
3. Start the server manually to confirm that the port number is available.

   See the installation documentation for your platform and the *Utility Guide*.

36.25 Investigating the Port Assignment

Follow these steps to find out if the port is in use.

Procedure

1. Look in the interfaces file to identify the port number assigned to the server.
2. Determine whether another process is using the same port number by entering:

   ```
   netstat -a
   ```

   If the port number is presented as a local address in the netstat output, you cannot use that port for the server, because another process is already using that port.
3. To verify that the server port is in use, start the server manually. The server does not start if its assigned port number is already in use.

For information on starting servers manually, see the installation documentation for your platform and the *SAP ASE Utility Guide*.

36.26 If a Stale Server Process Is Retaining Use of the Port Number

Follow one of the two options, and then start the server manually to confirm that the port number is available.

- Use the operating system kill command to terminate the process.
- Use another port number for the server by modifying the interfaces file.

For information on starting servers manually, see the SAP ASE installation guide for your platform and the SAP ASE Utility Guide.

36.27 Error when Executing an ESP

If you receive an error when executing an extended stored procedure (ESP), check for processes that might be using the same port number.

Use the `netstat` command to determine if the port number specified for XP Server is in use.

If you attempt to execute an ESP (extended stored procedure), you may see:

```
00:00000:00008:1997/09/10 12:52:53.03 kernel  XP Server failed to start.
     Try bringing up XP Server manually. Check SQL Server documentation for more
     information on how to bring XP Server up.
```

- If you find no processes using the same port number:
  1. Restart SAP ASE.
  2. Execute the ESP that you attempted earlier. XP Server should start automatically.
- If you find a process using the same port number, you can:
  1. Change the interfaces file to use a new port number for the XP Server, or stop the process using the port number allotted to XP Server.
  2. Restart SAP ASE, and execute the ESP that you attempted earlier. XP Server should start automatically.

37 Use the Lightweight Directory Access Protocol as a Directory Service

Lightweight Directory Access Protocol (LDAP) is an industry standard for accessing directory services. Directory services allow components to look up information by a distinguished name (DN) from an LDAP server that stores and manages server, user, and software information that is used throughout the enterprise or over a network.

SAP ASE uses directory services to establish client and RPC connections over the Internet. LDAP directory services are used to establish connections. LDAP is a licensed feature of SAP ASE.

The LDAP server can be located on a different platform from the one on which SAP ASE or the clients are running. LDAP defines the communication protocol and the contents of messages exchanged between clients and servers. Messages are operators, such as client requests for read, write and query, and server responses, including data-format information.

The LDAP server stores and retrieves information about:

- SAP ASE, such as IP address, port number, and network protocol
- Security mechanisms and filters
- High availability companion server name

You can Configure the LDAP server with these access restrictions:

- Anonymous authentication – all data is visible to any user.
- User name and password authentication – SAP ASE uses the default user name and password for UNIX platforms:
  - `$SYBASE/$SYBASE_OCS/config/libtcl.cfg` on 32-bit platforms
  - `$SYBASE/$SYBASE_OCS/config/libtcl64.cfg` on 64-bit platforms

User name and password authentication properties establish and end a session connection to an LDAP server.

> **Note**
>
> The user name and password that are passed to the LDAP server for user authentication purposes are distinct and different from those used to access SAP ASE.

When an LDAP server is specified in the `libtcl.cfg or libtcl64.cfg` file (collectively the `libtcl*.cfg` file), the server information is accessible only from the LDAP server. SAP ASE ignores the interfaces file.

For the SAP ASE Cluster Edition, an interfaces file may be set in the quorum file. When the quorum file specifies an interfaces file, the SAP ASE Cluster Edition ignores the directory services specified in `libtcl*.cfg` files.

If multiple directory services are supported in a server, the order in which they are searched is specified in `libtcl*.cfg`. You cannot specify the search order with the `dataserver` command line option.

See the *Security Administration Guide*.

37.1 LDAP Directory Services Versus the SAP Interfaces File

The LDAP driver implements directory services for use with an LDAP server.

LDAP directories provide:

- A network-based alternative to the traditional SAP interfaces file
- A single, hierarchical view of information, including users, software, resources, networks, files, and so on

interfaces File	Directory Services
Platform-specific	Platform-independent
Specific to each SAP installation	Centralized and hierarchical
Contains separate master and query entries	One entry for each server that is accessed by both clients and servers
Cannot store metadata about the server	Stores metadata about the server

LDAP directory services support more attributes than the SAP interfaces file. These attributes can include server version, server status, and so on.

> **Note**
>
> LDAP is supported only with reentrant libraries. Use `isql_r` instead of `isql`.

Table 12: SAP LDAP Directory Definitions

Attribute name	Value type	Description
ditbase	`interfaces` file or `libtcl.cfg`	DIT base for object tree. If the `libtcl.cfg` file is specified, the `interfaces` file is ignored. You can use `ct_con_prop`() to override the `libtcl.cfg` file for a specified connection.
dn	Character string	Distinguished name. Must be a unique name that identifies the object.
sybaseVersion	Integer	Server version number.
sybaseServername	Character string	Server name.
sybaseService	Character string	Service type: SAP ASE
sybaseStatus	Integer	Status: 1 = Active, 2 = Stopped, 3 = Failed, 4 = Unknown.
sybaseAddress	String	Each server address includes: • Protocol: TCP, NAMEPIPE, SPX DECNET (entry is case sensitive). • Address: any valid address for the protocol type. **Note** `dscp` splits this attribute into Transport type and Transport address. • Filter: None, ssl, or ssl="CN=common_name". `dscp` splits this attribute into Transport type and Transport address.
sybaseSecurity (optional)	String	Security OID (object ID).

Attribute name	Value type	Description
sybaseRetryCount	Integer	This attribute is mapped to CS_RETRY_COUNT, which specifies the number of times that `ct_connect` retries the sequence of network addresses associated with a server name.
sybaseRetryDelay	Integer	This attribute is mapped to CS_LOOP_DELAY, which specifies the delay, in seconds, that `ct_connect` waits before retrying the entire sequence of addresses.
sybaseHAservername (optional)	String	A secondary server for failover protection.

The traditional interfaces file with TCP connection and a failover machine looks like:

```
master tcp ether huey 5000
query tcp ether huey 5000
hafailover secondary
```

An example of an LDAP entry with TCP and a failover machine looks like:

```
dn: sybaseServername=foobar, dc=sybase,dc=com
objectClass: sybaseServer
sybaseVersion: 1500
sybaseServername: foobar
sybaseService: ASE
sybaseStatus: 4
sybaseAddress: TCP#1#foobar 5000
sybaseRetryCount: 12
sybaseRetryDelay: 30
sybaseHAServernam: secondary
```

All entries in the LDAP directory service are called entities. Each entity has a distinguished name (DN) and is stored in a hierarchical tree structure based on its DN. This tree is call the directory information tree (DIT). Client applications use a DIT base to specify where entities are stored.

In the example above, the entry describes an SAP ASE named "foobar" listening on a TCP connection with a port number of 5000. This entity also specifies a retry count of 12 (times) and a retry delay of 30 (seconds). Once a client has found an address where a server responds, the login dialog between the client and the server begins.

You can find a complete list of the SAP LDAP directory schema in UNIX in a file called `sybase.schema` in the directory `$SYBASE/$SYBASE_OCS/config`. In the same directory, there is also a file called `sybase-schema.conf`, which contains the same schema, but uses a Netscape-specific syntax.

Since LDAP supports multiple entries for each attribute, each address attribute must contain the address of a single server, including protocol, access type, and address.

For example, this is an LDAP entry for a Windows server listening on two addresses, with different connection protocols:

```
sybaseAddress = TCP#1#TOEJAM 4444
sybaseAddress = NAMEPIPE#1#\pipe\sybase\query
```

> **Note**
>
> Each entry in the address field is separated by the # character.

You can edit these entries with `dsedit`.

To ensure cross-platform compatibility for all SAP products, the protocol and address attribute fields should be in a platform- and product-independent format.

37.2 The libtcl*.cfg File

Use the `libtcl*.cfg` file to specify the LDAP server name, port number, DIT base, user name, and password to authenticate the connection to an LDAP server.

The purpose of the `libtcl*.cfg` file is to provide configuration information such as driver, directory, and security services for Open Client/Open Server and Open Client/Open Server-based applications. 32-bit utilities, such as `dsedit` and `srvbuild`, look up the `libtcl.cfg`, while 64-bit applications use the `libtcl64.cfg` file for configuration information.

Edit both the `libtcl.cfg` and the `libtcl64.cfg` files to ensure compatibility between 32- and 64-bit applications.

The default `libtcl.cfg` file is located in `$SYBASE/$SYBASE_OCS/config`.

If LDAP is specified in the `libtcl.cfg` file, the interfaces file is not used.

> **Note**
>
> Open Client/Open Server applications that use the `-I` option at start-up override the `libtcl.cfg` file and use the interfaces file.

In its simplest form, the `libtcl.cfg` file is in this format:

```
[DIRECTORY]
```

```
ldap=libsybdldap.dll  ldapurl
```

where the `<ldapurl>` is defined as:

```
ldap://<host:port/ditbase>
```

This LDAP entry, using these same attributes, is an anonymous connection and works only if the LDAP server allows read-only access.

```
ldap=libsybdldap.dll ldap://ldaphost/d=sybase,dc=com
```

To enable password authentication at connection time, you can specify a user name and password in the `libtcl.cfg` file as extensions to the LDAP URL.

37.3 Enabling LDAP Directory Services

Enable LDAP directory service.

Procedure

1. Install the "Security and directory services" license package ASE_SECDIR.
2. Add the location of the LDAP libraries to the UNIX load library path environment variable for your platform.
3. Configure the `libtcl.cfg` file to use directory services.

   Use any standard ASCII text editor to:

   - Remove the semicolon (;) comment markers from the beginning of the LDAP URL lines in the `libtcl.cfg` file under the `[DIRECTORY]` entry.
   - Add the LDAP URL under the `[DIRECTORY]` entry.

   > ⚠ Caution
   >
   > The LDAP URL must be on a single line.

   ```
   libtcl.cfg
   ldap=libsybdldap.so ldap://host:port/ditbase??scope??bindname=username
   password
   libtcl64.cfg
   ldap=libsybdldap64.so ldap://host:port/ditbase??scope??bindname=username
   password
   ```

   For example:

   ```
   [DIRECTORY]
   ldap=libsybdldap.so ldap:///huey:11389/dc=sybase,dc=com??one??
   bindname=cn=Manager,dc=sybase,dc=com secret
   ```

   "*one*" indicates the scope of a search that retrieves entries one level below the DIT base.
4. Verify that the appropriate environment variable points to the required third-party libraries. The Netscape LDAP SDK libraries are located in `$SYBASE/$SYBASE_OCS/lib3p` or `lib3p64`.The UNIX load library path environment variable must point to this directory.
5. Add your server entry to the LDAP server using `dscp` or `dsedit`.

37.4 Keywords for the LDAP URL Variables

Configure the `libtcl.cfg` file `DIRECTORY` section.

Keyword	Description	Default
`<host>` (required)	The host name or IP address of the machine running the LDAP server	None
`<port>`	The port number that the LDAP server is listening on	389
`<ditbase>` (required)	The default DIT base	None
`<username>`	Distinguished name (DN) of the user to authenticate	NULL (anonymous authentication)
`<password>`	Password of the user to be authenticated	NULL (anonymous authentication)

37.5 Adding a Server to the Directory Services

Use `dsedit` to add a new server entry.

Context

Each server entry is made up of a set of attributes. When you add or modify a server entry, you are prompted for information about server attributes. Some attributes are provided by default, others require user input. When a default value is provided, it appears in brackets "[ ]". See SAP LDAP directory definitions for accepted values.

Use `srvbuild` to add entries, but not modify or delete them.

> ⚠ Caution
>
> Most LDAP servers have an `ldapadd` utility for adding directory entries. SAP recommends you use `dsedit` instead since it has built-in semantic checks that generic tools do not provide.

Procedure

1. Source `SYBASE.csh` or `SYBASE.sh` to set the environment variables.
2. `cd` to `$SYBASE/$SYBASE_OCS/bin`.

3. Execute `dsedit`.
4. Select **LDAP** from the list of servers, and click **OK.**
5. Click **Add New Server Entry**.
6. Enter:
   - The server name – required.
   - The security mechanism – optional. This is the name of the high-availability failover server, if you have one.
7. Click **Add New Network Transport** and:
   - Select the transport type.
   - Enter the host name.
   - Enter the port number.
8. Click **OK** twice to exit `dsedit`.
   To view the server entries, enter this URL in your browser:

   `http://host:port/ditbase??one`.

   For example:

   `ldap://huey:11389/dc=sybase,dc=com??one`

   For more information about `dscp`, see the *Open Client/Server Configuration Guide*.

37.6 Multiple Directory Services

You can specify multiple directory services for high availability failover protection.

Not every directory service in the list must be an LDAP server. Any type of LDAP service, whether it is an actual server or a gateway to other LDAP services, is called an LDAP server.

For example:

```
[DIRECTORY]
```

```
ldap=libsybdladp.so ldap://test:389/dc=sybase,dc=com
ldap=libsybdldap.so ldap://huey:11389/dc=sybase,dc=com
```

In this example, if the connection to `<test:389>` fails, the connection fails over to the DCE driver with the specified DIT base. If this also fails, a connection to the LDAP server on `<huey:11389>` is attempted. Different vendors employ different DIT base formats.

> **Note**
>
> For more information, see the *Open Client Client-Library/C Programmer Guide* and the *Open Client Client-Library/C Reference Manual*

37.7 Encrypting the Password

Entries in the `libtcl.cfg` file are in human-readable format. SAP provides a `pwdcrypt` utility for basic password encryption.

Context

`pwdcrypt` is a simple algorithm that, when applied to keyboard input, generates an encrypted value that can be substituted for the password. `pwdcrypt` is located in `$SYBASE/$SYBASE_OCS/bin`.

Procedure

1. From the `$SYBASE/$SYBASE_OCS` directory, enter:

   ```
   bin/pwdcrypt
   ```

2. Enter your password twice when prompted.
   `pwdcrypt` generates an encrypted password. For example:

   ```
   0x01312a775ab9d5c71f99f05f7712d2cded2i8d0ae1ce78868d0e8669313d1bc4c706
   ```

3. Copy and paste the encrypted password into the `libtcl.cfg` file using any standard text editor. Before encryption, the file entry appears as:

   ```
   ldap=libsybdldap.so
   ldap://ldaphost/dc=sybase,dc=com??one??bindname=uid=Manager,dc=sybase,
   dc=com?password
   ```

4. Replace the password with the encrypted string:

   ```
   ldap=libsybdldap.so
   ldap://ldaphost/dc=sybase,dc=com??one??bindname=uid=Manager,dc=sybase,dc=com?
   0x01312a775ab9d5c71f99f05f7712d2cded2i8d0ae1ce78868d0e8669313d1bc4c706
   ```

   ⚠ Caution

   Even if your password is encrypted, you should still protect it using file-system security.

37.8 Performance When Using LDAP

Performance when using an LDAP server may be slower than when using an interfaces file because the LDAP server requires time to make a network connection and retrieve data.

Since this connection is made when SAP ASE is started, any performance degradation is seen at login, if at all. During normal system load, the delay should not be noticeable. During high system load with many

connections, especially repeated connections with short duration, the overall performance difference of using an LDAP server versus the traditional interfaces file might be noticeable.

37.9 Migrating from the Interfaces File to LDAP

There is no direct method to upgrade an existing server using the `interfaces` file to one that uses lightweight directory services. To upgrade an earlier version of SAP ASE to the newest version, see the installation guide for your platform. Once you have upgraded the server, you can configure your server to use LDAP service.

Procedure

1. Shut down the server.
2. Edit the `$SYBASE/$SYBASE_OCS/config/libtcl.cfg` or `libtcl64.cfg` file to add the directory service.
3. Add the server, cluster, and instance entries for clustered servers to the directory service using `dsedit` or `dscp`.
4. Verify that the interfaces directory attributes in the cluster quorum file is empty for the cluster and instance definitions using `qrmutil`.

   This example shows the values in the quorum file:

   ```
   $SYBASE/$SYBASE_ASE/bin/qrmutil --quorum-dev=<path_to_your_quorum> --
   display=config
   ```

   Reset the value if a path is defined for the `<interface_dir>` attribute on any instance, or for the cluster. Specifying this attribute with a path value forces the instance to use the interfaces file and overrides the information in the `libtcl.cfg` and `libtcl64.cfg` files.

   For example, use the `qrmutil` utility to reset the value of the `interface_dir` attributes The value for `<interface_dir>` is two single quotes, meaning an empty string.

   ```
   $SYBASE/$SYBASE_ASE/bin/qrmutil --quorum-dev=<path_to_your_quorum> --
   <interface_dir>=''
   $SYBASE/$SYBASE_ASE/bin/qrmutil --quorum-dev=<path_to_your_quorum> --
   instance=<name_of_instance_to_reconfig> --<interface_dir>=''
   ```
5. Restart your server or cluster.

Related Information

Adding a Server to the Directory Services [page 256]
Enabling LDAP Directory Services [page 255]

37.10 Using LDAP Directory Services with the Shared-Disk Cluster

The SAP ASE Cluster Edition can use LDAP directory services to specify its cluster and instance entries.

You must specify an empty string for the `<interface_dir>` attribute in the cluster's quorum file. Do not use the `dataserver` parameter `-i <interfaces_path>` to specify the path to the interfaces file. If you do not specify a value for `<interface_dir>`, the SAP ASE Cluster Edition uses the ordered list of directory services defined in the `$SYBASE/OCS-15_0/config/libtcl64.cfg` (for 64-bit servers and clients), or `$SYBASE/OCS-15_0/config/libtcl.cfg` (for 32-bit servers and clients). After the server searches the directory services defined in `libtcl64.cfg`, the interfaces file in the default location is searched.

Open Client applications can use LDAP directory service to store cluster and instance server entries. For example, for a cluster named "mycluster" with two instances ("ase1" and "ase2"), the interfaces looks similar to:

```
ase1     master tcp ether blade1 10945
     query tcp ether blade1 10945
ase2      master tcp ether blade2  10955
     query tcp ether blade2  10955
mycluster      query tcp ether blade1 10945
     query tcp ether blade2 10955
```

Use `dsedit` or `dscp` to add equivalent LDAP directory service entries for the server names "ase", "ase2", and "mycluster" to the LDAP directory service. For more information about `dsedit` and `dscp`, see the *SAP ASE Utility Guide*.

Clients can connect to any instance in the cluster using the cluster name (in this example, "mycluster") or an instance-specific server name ("ase1" or "ase2").

When SSL is used for clients to connect to a nonclustered SAP ASE using SSL, the SSL filter is placed after the port number in the interfaces file. The directory service includes the common name, which you added manually or with `dsedit`. Typically, one SSL certificate with one common name is used for the entire cluster, rather than one for each instance.

This example adds the SSL filter to an interfaces file entry for the cluster "mycluster":

```
mycluster
    query tcp ether blade1 10945 ssl="cn=mycluster.domain.com"
    query tcp ether blade2 10955 ssl="cn=mycluster.domain.com"
```

Entries added to an LDAP directory service must specify the common name with the SSL filter, `ssl="cn=mycluster.domain.com"`.

For example, this `dscp` session adds the example entry above for cluster "mycluster":

```
%  dscp
>> open ldap
ok
Session 1 ldap>> add mycluster
Service: [ASE]
Transport Type: [tcp]
Transport Address: blade1 10945 ssl="cn=mycluster.domain.com"
Transport Type: [tcp]
Transport Address: blade2 10955 ssl="cn=mycluster.domain.com"
Transport Type: [tcp]
```

```
Transport Address:
Security Mechanism [] :
HA Failoverserver:
Retry Count:
Retry Delay:
Added mycluster
Session 1 ldap>> read mycluster
DIT base for object:  dc=domain,dc=com
Distinguish name: sybaseServername=mycluster, dc=domain,dc=com
Server Entry Version: 15001
Server Name: mycluster
Server Service: ASE
Server Status: 4 (Unknown)
Server Address:
Transport Type: tcp
Transport Address: yellowstar 2521 ssl="cn=mycluster.domain.com"
Transport Type: tcp
Transport Address: yellowstar 2525 ssl="cn=mycluster.domain.com"
Session 1 ldap>> quit
```

Related Information

Adding a Server to the Directory Services [page 256]
Adding SSL Common Name [page 210]

38 Localization Support

SAP ASE supports localization for international customers and for customers with heterogeneous environments.

Localization is setting up an application to run in a particular language or country environment, including translated system messages and correct formats for date, time, and currency.

By default, the SAP ASE and Backup Server configurations use the English locale settings, which include:

- Character set definition files for Western European character sets
- Sort-order definition files for Western European character sets
- US English system message files

During installation or through reconfiguration, you can specify a different language, character set, and sort order.

SAP ASE supports localization for international customers and for customers with heterogeneous environments. This support includes:

- Data processing support – SAP ASE comes with character set and sort order definition files it uses to process the characters used in languages:
  - Western Europe
  - Eastern Europe
  - Middle East
  - Latin America
  - Asia
- Translated system messages – SAP ASE includes language modules for:
  - Brazilian Portuguese
  - Chinese (Simplified)
  - French
  - German
  - Japanese
  - Korean
  - Polish
  - Spanish
  - Thai

38.1 Language Modules

SAP ASE stores its localized software messages in separate language modules.

When you install a language module, the installation program loads the messages, character set, and sort-order files that support the new language in the correct locations.

When you install SAP ASE and Backup Server, system messages in English are installed by default.

38.2 Default Character Sets

The default character set is the one in which data is encoded and stored on the SAP ASE databases.

Change the Default Language and Character Set

> **Caution**
>
> Make all changes to the character set and sort order for a new SAP ASE before creating any user databases or making any changes to the SAP-supplied databases. Changing the character set and sort order after data or data structures have been added to SAP ASE may require additional, time-consuming steps. To change the character set or sort order after you have added data, see the *System Administration Guide: Volume 1*.

After `srvbuild` configures a new SAP ASE, you are asked whether to localize your SAP ASE to a language other than us_english and whether to use a character set or sort order other than the default.

The default SAP ASE uses:

- us_english language
- iso_1 character set
- Binary sort order

Change the Default Character Set for Servers

You can select any character set as the default on SAP ASE, including character sets that are not the platform default character sets. When selecting a new default character set:

- To avoid conversion errors or overhead, determine the default character set based on the character set used by your clients. For example, if most of your clients use ISO 8859-1, specify ISO 8859-1 to minimize the amount of data conversion.
- If your server is operating in a heterogeneous language environment, choose a character set that works with all the character sets needed. Often, this is Unicode (UTF-8).

> **Caution**
>
> Make all changes to the default character set and sort order for a new SAP ASE before creating any user databases or making any changes to the SAP-supplied databases. Changing the character set and sort order after data or data structures have been added to SAP ASE can cause incorrect behavior. To change the character set or sort order after you have added data, see the *System Administration Guide: Volume 1*.

38.3 Supported Character Sets

SAP ASE supports many languages, scripts and character sets.

38.3.1 Arabic Character Sets

SAP ASE supports Arabic character sets.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
cp864	X	PC Arabic
cp1256	X	Microsoft Windows Arabic
iso88596	X	ISO 8859-6 Latin/Arabic

38.3.2 Baltic Character Set

SAP ASE supports the Baltic character set.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
cp1257	X	Microsoft Windows Baltic

38.3.3 Simplified Chinese Character Sets

SAP ASE supports Simplified Chinese character sets.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
eucgb	X	EUC GB encoding = Simplified Chinese character sets

Character Set	Unilib Required	Description
cp936	X	Microsoft Simplified Chinese character sets
gb18030	X	RC 18030 standard

38.3.4 Traditional Chinese Character Set

SAP ASE supports Traditional Chinese character sets.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
cp950	X	PC (Microsoft) Traditional Chinese
euccns	X	EUC CNS encoding = Traditional Chinese with extensions
big5	X	Big 5 Traditional Chinese
big5hk	X	Big 5 with HKSCS extensions

38.3.5 Cyrillic Character Sets

SAP ASE supports Cyrillic character sets.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
cp855		IBM PC Cyrillic
cp866		PC Russian
cp1251		Microsoft Windows 3.1 Cyrillic
iso88595		ISO 8859-5 Latin/Cyrillic
koi8		KOI-8 Cyrillic
mac_cyr		Macintosh Cyrillic
kz1048		Kazakhstan Cyrillic

38.3.6 Eastern European Character Sets

SAP ASE supports Eastern European character sets.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
cp852		PC Eastern Europe
cp1250		Microsoft Windows 3.1 Eastern European
iso88592		ISO 8859-2 Latin-2
mac_ee		Macintosh Eastern European

38.3.7 Greek Character Sets

SAP ASE supports Greek character sets.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
cp869		IBM PC Greek
cp1253		MS Windows Greek
greek8		HP GREEK8
iso88597		ISO 8859-7 Latin/Greek
macgrk2		Macintosh Greek

38.3.8 Hebrew Character Sets

SAP ASE supports Hebrew character sets.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
cp1255	X	Microsoft Windows Hebrew

Character Set	Unilib Required	Description
iso88598	X	ISO 8859-8 Hebrew

38.3.9 Japanese Character Sets

SAP ASE supports Japanese character sets.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
cp932	X	IBM J-DBCS:CP897 + CP301 (Shift-JIS)
deckanji		Digital UNIX JIS encoding
eucjis		EUC-JIS encoding
sjis		Shift-JIS (no extensions)

38.3.10 Korean Character Set

SAP ASE supports the Korean character set.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
eucksc	X	EUC KSC Korean encoding = CP949

38.3.11 Thai Character Sets

SAP ASE supports Thai character sets.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
tis620	X	TIS-620 Thai standard

Character Set	Unilib Required	Description
cp874	X	Microsoft Windows Thai

38.3.12 Turkish Character Sets

SAP ASE supports Turkish character sets.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
cp857		IBM PC Turkish
cp1254		Microsoft Windows Turkish
iso88599		ISO 8859-9 Latin-5 Turkish
macturk		Macintosh Turkish
turkish8		HP TURKISH8

38.3.13 Unicode Character Set

SAP ASE supports the Unicode character set (which supports over 650 languages).

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
utf8	X	Unicode UTF-8 encoding

38.3.14 Vietnamese Character Set

SAP ASE supports Vietnamese character sets.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
cp1258	X	Microsoft Windows Vietnamese

38.3.15 Western European Character Sets

SAP ASE supports Western European character sets.

- X – requires Unilib® conversion.
- No X – may use either the Unilib conversion or the built-in conversion.

Character Set	Unilib Required	Description
ascii8	X	US ASCII, with 8-bit data, ISO 646
cp437		IBM CP437 – US code set
cp850		IBM CP850 – European code set
cp860	X	PC Portuguese
cp863	X	IBM PC Canadian French code page
cp1252	X	Microsoft Windows US (ANSI)
iso_1		ISO 8859-1 Latin-1
mac		Standard Macintosh coding
roman8		HP ROMAN8
iso 885915	X	ISO 8859-15 Latin-1 with Euro support

38.4 Character Set Names

Supported character sets and their SAP name.

Character Sets	SAP Name
ASCII 8	acsii_8
Big 5	big5
Big 5HK	big5hk

Character Sets	SAP Name
CP 437	cp437
CP 850	cp850
CP 852	cp852
CP 855	cp855
CP 857	cp857
CP 858	cp858
CP 860	cp860
CP 864	cp864
CP 866	cp866
CP 869	cp869
CP 874	cp874
CP 932	cp932
CP 936	cp936
CP 950	cp950
CP 1250	cp1250
CP 1251	cp1251
CP 1252	cp1252
CP 1253	cp1253
CP 1254	cp1254
CP 1255	cp1255
CP 1256	cp1256
CP 1257	cp1257
CP 1258	cp1258
DEC Kanji	deckanji
EUC-CNS	euccns
EUC-GB	eucgb

Character Sets	SAP Name
EUC-JIS	eucjis
EUC-KSC	eucksc
GB 18030	gb18030
GREEK8	greek8
ISO 8859-1	iso_1
ISO 8859-2	iso88592
ISO 8859-5	iso88595
ISO 8859-6	iso88596
ISO 8859-7	iso88597
ISO 8859-8	iso88598
ISO 8859-9	iso88599
ISO 8859-15	iso15
Kazakhstan Cyrillic	kz1048
Koi8	koi8
ROMAN8	roman8
ROMAN9	roman9
Shift-JIS	sjis
TIS 620	tis620
TURKISH8	turkish8
UTF-8	utf8

38.5 Load Character Sets

Use the `charset` utility to load character sets and sort orders into SAP ASE. Use `charset` to load the default character set and sort order only during installation.

To change the default character set and sort order of SAP ASE, see the *System Administration Guide: Volume 1*.

Syntax

```
charset
```

```
[ -U <username> ]
```

```
[ -P <password> ]
```

```
[ -S <server> ]
```

```
[ -I <interfaces> ]
```

```
[ -v <version> ]
```

```
<sort_order>
```

```
[ <charset> ]
```

Table 13: Keywords and Options for Charsets

Keywords and Options	Description
-U	If you are not already logged in to your operating system as "sa", specify "-Usa" or "/username = sa" in the command line.
-P	Specifies the "sa" password on the command line. If not specified, the user is prompted for the "sa" password.
-S	Specifies the name of the server. If not specified, `charset` uses the DSQUERY environment variable to identify the server name. If there is no DSQUERY environment variable, `charset` attempts to connect to a server named "SYBASE".
-I	Specifies the interfaces file to use. If not specified, `charset` uses the interfaces file in the SYBASE directory.
-v	Prints the SAP version, then exits. Do not specify other options with -v.
<sort_order>	When `charset` is used to load the default character set and sort order, <sort_order> is a required parameter that specifies the name of the sort order file that SAP ASE use. When loading additional character sets, use `charset.loc` to indicate the name of the character-set files.
<charset>	Specifies the directory of the character set that SAP ASE uses.

38.6 Language Definition Files Sort Order

Supported sort orders.

Languages not list listed do not have a language-specific sort order; use a binary sort order.

Table 14: Available Sort Orders

Language or Script	Sort Orders	File Name	ID
All languages	Binary order	`binary.srt`	50
Central European Czech, Slovak These sort orders work only with CP 852, CP 1250, and ISO 8859-2	Dictionary order, case sensitive, accent sensitive	czedit.srt	80
	Dictionary order, case insensitive, accent sensitive	`czeocs.srt`	82
	Dictionary order, case insensitive, accent insensitive	`czenoac.srt`	81
Cyrillic	Dictionary order, case sensitive, accent sensitive	`cyrdict.srt`	63
	Dictionary order, case insensitive, accent sensitive	`cyrnocs.srt`	64
English French German These sort orders work with all Western European character sets.	Dictionary order, case sensitive, accent sensitive	dictiona.srt	51
	Dictionary order, case insensitive, accent sensitive	`nocase.srt`	52
	Dictionary order, case insensitive, accent sensitive, with preference	`nocasepr.srt`	53
	Dictionary order, case insensitive, accent insensitive	`noaccent.srt`	54
English French German These sort orders work only with CP 850.	Alternate dictionary order, case sensitive	`altdict.srt`	45
	Alternate dictionary order, case sensitive, accent insensitive	`altnoacc.srt`	39
	Alternate dictionary order, case sensitive, with preference	`altnocsp.srt`	46
Greek This sort order works only with ISO 8859-7.	Dictionary order, case sensitive, accent sensitive	`elldict.srt`	65

Language or Script	Sort Orders	File Name	ID
Hungarian These sort orders work only with ISO 8859-2.	Dictionary order, case sensitive, accent sensitive	`hundict.srt`	69
	Dictionary order, case insensitive, accent sensitive	`hunnoac.srt`	70
	Dictionary order, case insensitive, accent insensitive	`hunnocs.srt`	71
Russian This sort order works with all Cyrillic character sets except CP 855.	Dictionary order, case sensitive, accent sensitive	`rusdict.srt`	58
	Dictionary order, case insensitive, accent sensitive	`rusnocs.srt`	59
Scandinavian These sort orders work only with CP 850.	Dictionary order, case sensitive, accent sensitive	`scandict.srt`	47
	Dictionary order, case insensitive, with preference	`scannocp.srt`	48
Spanish	Dictionary order, case sensitive, accent sensitive	`espdict.srt`	55
	Dictionary order, case insensitive, accent sensitive	`espnocs.srt`	56
	Dictionary order, case insensitive, accent insensitive	`espnoac.srt`	57
Thai	Dictionary order	`dictionary.srt`	51
Turkish These sort orders work only with ISO 8859-9.	Dictionary order, case sensitive, accent sensitive	`turdict.srt`	72
	Dictionary order, case insensitive, accent insensitive	`turnoac.srt`	73
	Dictionary order, case insensitive, accent sensitive	`turnocs.srt`	74

38.7 Character Set Conversion

Backup Server passes messages to SAP ASE in the client's language and in the SAP ASE character set. SAP ASE converts the messages and issues them in the client's language and character set.

When selecting a character set:

- In a heterogeneous environment, SAP ASE and Backup Server may need to communicate with clients running on different platforms and using different character sets. To maintain data integrity, the server converts the code between the character sets.
- To use the built-in conversion, install the character set definition files on the server for all the character sets being used by your clients. Built-in conversion support is available for many character sets.
- Unilib conversion support is available for all character sets supported by SAP. To enable Unilib conversion, you must use `sp_configure` and turn `enable unicode conversions` on. See the *System Administration Guide: Volume 1*.

If either SAP ASE or Backup Server does not support a client's language or character set, the server issues a warning message. Errors also occur when the Backup Server character set is incompatible with the SAP ASE character set. By default, Unicode conversion is enabled.

Character set conversion is supported only between character sets for the same language or between character sets in the same language group.

For example, automatic character set conversion is supported between the character sets for the Western European languages: ASCII 8, CP 437, CP 850, CP 860, CP 863, CP 1252, ISO 8859-1, ISO 8859-15, and ROMAN8. Similarly, conversion is supported between the character sets for Japanese: CP 932, EUC-JIS, Shift-JIS, and DEC-Kanji.

However, code conversion is not supported between any of the Western European language character sets and the Japanese character sets.

For more information about supported conversions, see the *System Administration Guide: Volume 1*.

38.7.1 Conversions Between Server and Client

If SAP ASE does not support the client's language or character set, the client can connect with the server, but no character conversions occur.

When a localized client application connects to SAP ASE, the server checks to see if it supports the client's language and character set.

- If SAP ASE supports the language, it automatically performs all character-set conversions and displays its messages in the client's language and character set.
- If SAP ASE does not support the language, it uses the user's default language or the default language of the SAP ASE.
- If SAP ASE does not support the character set, it issues a warning to the client, turns conversion off, and sets the language to US English.

38.8 Sort Orders

Each character set comes with one or more sort orders (collating sequences), which are located in the sort-order definition files (`.srt` files). These files accompany the character-set definition files and are in the same directory.

Although you can select a sort order for your data according to the needs at your site, the server can support only one sort order at a time; select one that works for all of your clients.

> **Caution**
>
> Make all changes to the default character set and sort order for a new SAP ASE before creating any user databases or making any changes to the SAP-supplied databases. Changing the character set and sort order after data or data structures have been added to SAP ASE may cause incorrect behavior. To change the character set or sort order after you have added data, see the *System Administration Guide: Volume 1*.

38.8.1 Available Sort Orders for Character Sets

The sort order determines the collating sequence SAP ASE uses to order, compare, and index character data. Each character set comes with one or more sort orders.

Sort orders are located in sort order definition files (`.srt` files) that accompany your character set definition files.

> **Note**
>
> Available sort orders vary according to the character set installed on SAP ASE.

You can see the available sort orders for your character set by looking in the `.srt` file for your language. Sort orders are stored in:

```
$SYBASE/charsets/<<charset_name>/>*.srt
```

Sort Order Name	Description
Binary order	Sorts all data according to numeric byte values for that character set. Binary order sorts all ASCII uppercase letters before lowercase letters. Accented or ideographic (multibyte) characters sort in their respective standards order, which may be arbitrary. All character sets have binary order as the default. If binary order does not meet your needs, you can specify one of the other sort orders either at installation or at a later time by, using the `charset` utility.
Dictionary order, case-sensitive, accent-sensitive	Sorts each uppercase letter before its lowercase counterpart, including accented characters. Recognizes the various accented forms of a letter and sorts them after the associated unaccented letter.

Sort Order Name	Description
Dictionary order, case-insensitive, accent-sensitive	Uppercase letters are equivalent to their lowercase counterparts and are intermingled in sorting results.
Dictionary order, case-insensitive, accent-insensitive	Diacritical marks are ignored.
Dictionary order, case-insensitive with preference	A word written with uppercase letters is equivalent to the same word written with lowercase letters. Uppercase and lowercase letters are distinguished only when you use an `order by` clause. The `order by` clause sorts uppercase letters before it sorts lowercase. Do not select this sort order unless your installation requires that uppercase letters be sorted before lowercase letters in otherwise equivalent strings for `order by` clauses. Using this sort order may reduce performance in large tables when the columns specified in an `order by` clause match the key of the table's clustered index.
Alternate dictionary order, case-sensitive	Use with several of the Western European languages.
Alternate dictionary order, case-insensitive, accent-insensitive	Use with several of the Western European languages.
Alternate dictionary order, case-insensitive, uppercase preference	Use with several of the Western European languages.
Spanish dictionary order, case-sensitive	Use with Spanish and for most Latin American locales.
Spanish dictionary order, case-insensitive	Use with Spanish and for most Latin American locales.
Spanish dictionary order case-insensitive, accent-insensitive	Use with Spanish and for most Latin American locales.
Scandinavian dictionary order, case-sensitive	Case-sensitive dictionary sort order. Use with Scandinavian languages.
Scandinavian dictionary order, case-insensitive, uppercase preference	Use with Scandinavian languages.

Use the `charset` utility to display the sort orders for the character sets you plan to use.

38.8.2 Default Unicode Sort Orders

The following default Unicode sort orders are available in ASE Cluster Edition:

Table 15: Default Unicode Sort Orders

Name	ID	Description
defaultml	20	Default Unicode multi-lingual ordering
thaidict	21	Thai dictionary ordering
iso14651	22	Ordering as per ISO14651 standard
utf8bin	24	Ordering for UTF-16 that matches the UTF-8 binary
binary	25	Binary sort
altnoacc	39	Alternate accent-insensitive
altdict	45	Alternate dictionary ordering
altnocsp	46	Alternate case-insensitive with preference
scandict	47	Scandinavian dictionary ordering
scannocp	48	Scandinavian case-insensitive with preference
bin_utf8	50	UTF-8 binary sort order
dict	51	General-purpose dictionary ordering
nocase	52	General-purpose case-insensitive dictionary ordering
nocasep	53	General-purpose case-insensitive with preference
noaccent	54	General-purpose accent-insensitive dictionary ordering
espdict	55	Spanish dictionary ordering
espnocs	56	Spanish case-insensitive dictionary ordering
espnoac	57	Spanish accent-insensitive dictionary ordering
rusnocs	59	Russian case-insensitive dictionary ordering
cyrnocs	64	Cyrillic case-insensitive dictionary ordering
elldict	65	Greek dictionary ordering
hundict	69	Hungarian dictionary ordering
hunnoac	70	Hungarian accent-insensitive dictionary ordering

Name	ID	Description
hunnocs	71	Hungarian case-insensitive dictionary ordering
turknoac	73	Turkish accent-insensitive dictionary ordering
cp932bin	129	CP932 binary sort order
dynix	130	GB pinyin sortorder
gb2312bn	137	GB2312 binary sort order
cyrdict	140	Cyrillic dictionary sort order
turdict	155	Turkish dictionary sort order
euckscbn	161	EUCKSC binary sort order
gbpinyin	163	GB pinyin sort order
rusdict	165	Russian dictionary sort order
sjisbin	179	SJIS binary sort order
eucjisbn	192	EUCJIS binary sort order
big5bin	194	BIG5 binary sort order

To view available Unicode sort orders, run `sp_helpsort`. For more information on unicode sort orders for UTF-8, see the *SAP ASE System Administration Guide*.

Related Information

Localization Directories [page 280]

38.9 Language Modules

SAP ASE stores its localized software messages in separate language module.

When you install a language module, the installation program loads the messages, character set, and sort-order files that support the new language in the correct locations.

By default, SAP ASE and Backup Server installs English system messages in English are installed by default.

38.9.1 Installing a New Language Module

A full installation of SAP ASE installs all the language components automatically. If you did not select a full installation, manually install additional language modules as required.

Procedure

1. Load the language module software from the distribution media. Load this software into the same directory in which you loaded SAP ASE.
2. Reconfigure the language and, if necessary, the character set and sort order for SAP ASE.

38.9.2 Message Languages

By default, the SAP ASE installs messages in U.S. English.

These rules apply to language modules:

- To specify a default language other than U.S. English, install the language module for the language you specify before you install or reconfigure the SAP ASE.
- If your clients require SAP ASE messages in a language other than US English, load the language module for those languages. before you configure SAP ASE to the language used by your clients.
- If SAP ASE does not support messages in a client's language, these clients receive messages in the server default language. For example, if your client's language is Latin, the Spanish language module is installed, and if Spanish is specified as the SAP ASE default language, the client receives messages in Spanish.

38.10 Localization Directories

SAP localization configuration involves the `locales` and `charsets` directories.

The table illustrates the structure of the localization files. It does not show a complete list of all the files.

`%SYBASE%/` or `$SYBASE/`	`charsets`	`charset_name`	`*.srt` files
		`charset_name...`	`charset.loc`
		`unicode`	`*.uct` files
	`locales`	`language_name`	`charset_name`
		`language_name...`	`charset_name...`

		locales.dat	--
		message	language_name
			language_name...

38.10.1 About the locales Directory

The `$SYBASE/locales` directory contains a subdirectory for each available language. Each language subdirectory contains a subdirectory for each available character set for that language.

- The `.loc` files in these subdirectories enable SAP ASE or Backup Server to report errors in a specific language, encoded in a specific character set. There are a variety of `.loc` files in each subdirectory, most of which contain translated error messages for a specific product or utility.
- The `common.loc` file in the `utf8` subdirectory for each language contains localized information used by all products, such as local date, time, and currency formatting.
- The `locales.dat` file contains entries that associate platform-specific locale names with SAP language and character-set combinations.

38.10.2 About the charsets Directory

The files in `$SYBASE/charsets/charset_name` contain information related to each particular character set, such as the definition of any available sort order.

38.10.3 Format of locales.dat File Entries

You can edit the `locales.dat` file to change the default language or character set for a platform, or to add new associations between platform locale names and SAP language and character set names.

Each entry in the `locales.dat` file links a platform-specific locale definition to a SAP language and character set combination and uses the following format:

```
locale = <platform_locale>, <syb_language>, <syb_charset>
```

where:

- `<platform_locale>` is the platform-specific keyword for a locale. For acceptable values, see your operating system documentation.
  When the locale being defined as the default for the site, `<platform_locale>` is “default.”
- `<syb_language>` is the name of the language directory to be used from within `$SYBASE/locales/language_name`.

- <syb_charset> is the character set that determines the character set conversion method and identifies the directory location of the message files for clients from within `$SYBASE/locales/language_name/charset_name`.

For example, this entry specifies that the default locale uses us_english for the language and iso_1 for the character set:

```
locale = default, us_english, iso_1
```

38.10.4 How Client Applications Use locales.dat

Client applications use the `locales.dat` file to identify which language and character set to use.

1. When a client application starts, it checks the operating system locale setting and the `locales.dat` file to see if the setting is appropriate for SAP ASE. For example, a locale entry for French looks like:

   ```
   locale = fr_FR, french, iso_1
   ```

2. When the client connects to SAP ASE, the language and character set information is passed to SAP ASE in the login record.
3. SAP ASE then uses:
   - The character-set information, for example, iso_1, to identify the client's character set and verify whether it can convert character data to this character set
   - The language (in the preceding example, French) and character set information to see if it has messages in the client's language.

> **Note**
>
> SAP ASE software includes some locale entries already defined in the `locales.dat` file. If these entries do not meet your needs, you can either modify them or add new locale entries.

38.11 Changing Localization Configuration

Each language uses about 2 MB of database space per module. If necessary, use the `alter database` command to increase the size of the `master` database before adding another language.

Context

> **Note**
>
> If you want to install more than one language on the SAP ASE Cluster Edition, and the `master` database is not large enough to manage more than one language, the transaction log may become too full. You can expand the `master` database only on the master device. See the *SAP ASE System Administration Guide*.

Procedure

1. Source `SYBASE.csh` or `SYBASE.sh` if you have not set up the Sybase environment variables.
2. Use the `langinstall` utility to configure localization for the SAP ASE Cluster Edition:

   ```
   $SYBASE/$SYBASE_ASE/bin/langinstall
   ```

   The syntax for `langinstall` is:

   ```
   langinstall [-S <server_name>] [-U <user_name>] [-P <password>]
   [-R <release_number>] [-I <path_to_interfaces>] [-v] <language>
   <character_set>
   ```

   This example installs the French language with the iso_1 default character set on `testserver`:

   ```
   langinstall -Usa -P -Smyserver french iso_1
   ```

38.11.1 Configuring SAP ASE for Other Character Sets

Configure SAP ASE with the character set and sort order for your language.

Prerequisites

The server must be running and you must have system administrator privileges.

Procedure

1. Use the `charset` utility to load the default character set and sort order.

   ```
   $SYBASE/$SYBASE_ASE/bin/charset -Usa -P<password>
   -S<server_name> <sort_order_file> <character_set>
   ```

   Replace `<sort_order_file>` with the name of the sort order file. Replace `<character_set>` with the SAP name for your character set.
2. Use `charset` utility to load any additional character sets.

   To use the SAP ASE built-in character-set conversions, you must load the character-set definition files for all the characters set on your client platforms. You do not need to do this if you are using the Unilib character-set conversions.
3. Using `isql`, log in to your server as "sa" and select the `master` database:

   ```
   1> use master
   2> go
   ```

4. Configure your server for the new character set and sort order:

```
1> sp_configure "default <sortorder_id>",
2> <sort_order_id>, "<character_set>"
3> go
```

   Replace `<sort_order_id>` with the ID for your sort order. Replace `<character_set>` with the SAP name for your character set.

5. Shut down the cluster. You can use `sybcluster`, the SAP Control Center, or, if you configured the cluster manually, a command line option.
6. Restart any instance in the cluster using `sybcluster` or the SAP Control Center.

   Note

   If you configured the cluster manually, use your normal process on UNIX systems to restart the instance. Typically, this means invoking the `dataserver` command:

```
$SYBASE/$SYBASE_ASE/BIN/dataserver
     --quorum_dev=<quorum_path> --instance=<instance_name>
```

7. The instance starts, rebuilds all the system indexes, then shuts down. Restart it a second time to bring the instance up in a stable state.
8. Check the cluster log file to verify that the character set and sort order changes have completed successfully.

38.11.2 Editing the locales.dat File

Use a text editor to make edits to the `locales.dat` file.

Prerequisites

Make a copy of the original file, in case you have problems with the resulting edited version.

Procedure

1. Using a text editor such as Notepad, open the `locales.dat` file copy.
2. Verify that the section contains an entry for the language (`<syb_language>`) and character set (`<syb_charset>`) combination you want to use.

   Note

   The value for `<platform_locale>` must match the value required by your operating system. If the locales definitions in your system configuration files do not match the SAP locale definitions, your applications do not run properly.

For example, for Open Client messages to appear in French, when SAP ASE is using the Roman8 character set, check the `locales.dat` entries for your platform and look for:

```
locale = fr_FR, french, roman8
```

3. Add the required entry or modify an existing entry.
4. Save the changes, if any, and exit the text editor.

38.11.3 Changing Backup Server Localization Configuration

Localize Backup Server by modifying the `RUN_<backup_server_name>` file.

Context

See the *Utility Guide* for more information on the `backupserver` utility.

39 SAP ASE Error Logging

Upon start-up, SAP ASE writes information to a local error log file, called the SAP ASE error log.

Logging to the SAP ASE error log is always enabled. However, when you create or modify a specific user-defined message, you can set it to be omitted from the log.

The error log file, `$SYBASE/$SYBASE_ASE/install/server_name.log`:

- Stores information about the success or failure of each start-up attempt
- Logs error and informational messages generated by the server during its operations
- Remains open until you stop the server process
- Contains start-up messages from SAP ASE

The SAP ASE Cluster Edition includes the instance ID in the front of the error log header with this format:

```
instance id: engine number : family id: process id: date time
```

If the error log become too large, you can:

- Use `sp_errorlog` to dynamically change its path. Once the older error log is not being used by SAP ASE, you can move it, and make space available.
- Stop the instance and delete logged messages.

See *Diagnosing System Problems* in the *System Administration Guide* for a description of the error log format.

39.1 Set Error Log Paths

The installation program sets the error log location in the SAP installation directory when you configure a new SAP ASE. Backup Server has its own error log.

The default location for each server's error log is:

- SAP ASE: `$SYBASE/$SYBASE_ASE/install/servername.log`
- Backup Server: `$SYBASE/$SYBASE_ASE/install/servername_back.log`

At start-up, you can reset the name and location of the SAP ASE error log file from the command line. Use the `-e` start-up parameter and value in the `dataserver` command to start SAP ASE.

However, if you are using `sybcluster` to manage the cluster, you must use the `sybcluster 'set instance logpath'` parameter to change the location of the error log file for each instance.

> **Note**
>
> Multiple instances cannot share the same error log.

39.1.1 Change Error Log Path

You can add a new location for the error log.

> **Note**
>
> The SAP ASE Cluster Edition installer does not create `RUN_server` files. However, if you create a `RUN_server` file, you can add a new location for the error log. If you do not provide a path to the error log, the SAP ASE Cluster Edition adds an error log according to the:
>
> - Cluster input file error log location. This information is stored in the quorum device and used by the `dataserver` utility.
> - Default location for error logs. If you do not supply an error log path, a log file named `instance_name.log` is created in the current working directory for `dataserver`.

You can change the log location stored on the quorum device using the `qrmutil --errorlog` parameter or with `sybcluster 'set instance errorlog'`.

If you are starting the cluster instances using shell scripts, change the value of the `dataserver -e` parameter, which overrides other settings for the error log location.

39.2 Manage Messages

Manage event logging by controlling whether a message is logged, or by adding user messages.

When event logging is enabled, you can:

- Use `sp_addmessage` to add a user message, or `sp_altermessage` to control whether a specific message is logged in the SAP ASE error log.
  See the *Reference Manual: Procedures*.
- Use configuration parameters to specify whether auditing events are logged. Auditing events pertain to a user's success (`log audit logon success`) or failure (`log audit logon failure`) in logging in to SAP ASE.

39.2.1 Log User-Defined Messages

You can write a user-defined messages to the SAP ASE error log.

SAP ASE lets you manage:

- New messages – include the `with_log` option in `sp_addmessage` when you add a new user-defined message to `sysusermessages`. This parameter sets the SAP ASE to log the message each time that the message appears.
- Existing messages – include the `with_log` option in `sp_altermessage` to change an existing user-defined message. This parameter alters the reporting status of that message:
  - TRUE – to enable logging.

- FALSE – to disable logging.

See `sp_addmessage` and `sp_altermessage` in the *Reference Manual: Procedures*.

39.2.2 Log Auditing Events

Use `sp_configure` parameters to specify whether SAP ASE writes auditing events, such as logins, to the SAP ASE error log.

By default, SAP ASE does not log auditing events. To log events, you can use `sp_configure` with these possible parameters and values:

- To enable logging of successful SAP ASE logins:

```
sp_configure "log audit logon success", 1
```

- To enable logging of unsuccessful SAP ASE logins:

```
sp_configure "log audit logon failure", 1
```

- To disable logging of that message type:

```
sp_configure "log audit logon success", 0
sp_configure "log audit logon failure", 0
```

See the *System Administration Guide: Volume 1*.

40 Adding Auditing

Auditing is an important part of security in a database management system. Security-related system activity is recorded in an audit trail, which you can use to detect penetration of the system and misuse of resources.

By examining the audit trail, the System Security Officer can inspect patterns of access to objects in databases and can monitor the activity of specific users. Audit records can be traced to specific users, enabling the audit system to act as a deterrent to users who are attempting to misuse the system.

A System Security Officer manages the audit system and is the only user who can start and stop auditing, set up auditing options, and process audit data.

40.1 Audit System Devices and Databases

The audit system includes several components.

The main components are:

- The `sybsecurity` device and the `sybsecurity` database, which stores audit information.
- The audit trail, which consists of several audit devices and tables that you determine at configuration time.
- The syslogs transaction log device, which stores transaction logs.

The sybsecurity Device and Database

The `sybsecurity` device stores the `sybsecurity` database. The `sybsecurity` database is created as part of the auditing configuration process. It contains all the system tables in the `model` database, as well as a system table for keeping track of server-wide auditing options and system tables for the audit trail.

Tables and Devices for the Audit Trail

The SAP ASE stores the audit trail in system tables, named `sysaudits_01` through `sysaudits_08`. At any given time, only one of the audit tables is current. The SAP ASE writes all audit data to the current audit table. A System Security Officer can use `sp_configure` to set or change which audit table is current.

When you configure the SAP ASE for auditing, you determine the number of audit tables for your installation. You can specify up to eight system tables (`sysaudits_01` through `sysaudits_08`). Plan to use at least two or three system tables for the audit trail and to put each system table on its own device, separate from the master device. If you do this, you can use a threshold procedure that archives the current audit table automatically, before it fills up and switches to a new, empty table for subsequent audit records.

Device for syslogs System Table

During auditing configuration, specify a separate device for the `syslogs` system table. The `syslogs` table, which exists in every database, contains a log of transactions that are executed in the database.

40.2 Running auditinit with the SAP ASE Cluster Edition

If you use `auditinit` with a nonclustered versions of SAP ASE, `auditinit` utility starts the server if it is not already running. However the SAP ASE Cluster Edition requires that an instance is running before you use `auditinit`.

`auditinit` does not start the instance if it has not already started. If you attempt to log into an SAP ASE Cluster Edition with `auditinit` that is not started, `auditinit` displays this warning message:

```
Can not login to server because it is not running.
Please manually start the server and retry
```

40.3 Preinstallation Tasks for Auditing Devices

Determine the location of the raw devices for the `sybsecurity`, `syslogs`, and `sysaudits` table devices. You will need to provide this information later.

We recommend that you:

- Configure your system with the minimum number of auditing devices you require - you must configure at least three devices. You can add more auditing devices later with `sp_addauditttable`. For information, see the *Reference Manual: Procedures*.
- Install auditing tables and devices in a one-to-one ratio. Tables that share the same device will share the same upper threshold limit. These tables cannot be used sequentially when a device fills up, because they both reside on the same device.
- Install each auditing table on its own device. This enables you to set up a smoothly running auditing system with no loss of auditing records. With two auditing tables, when one fills up, you can switch to the other. With a third auditing table, if one device fails, the System Security Officer can install a new threshold procedure that changes the device rotation to skip the broken device until the device is repaired.
- Make the device larger than the table. When you use only three auditing tables and devices, the size of the table and the size of the device can be similar, because you can obtain more auditing capacity by adding more auditing tables and devices (up to eight). When you are working toward the upper table and device limit (six to eight), you may want to make the device considerably larger than the table. Then, you can expand the table size later towards the upper size of the device when a larger auditing capacity is desired, and few or no device additions are available.

40.4 Installing Auditing

Installing auditing involves tasks such as configuring SAP ASE Cluster Edition, creating a device for an audit table, creating a database for the audit database transaction log, enabling auditing, and deleting and changing a device entry.

40.4.1 Configuring the SAP ASE Cluster Edition for Auditing

Follow the configuration steps and then restart the SAP ASE Cluster Edition.

Procedure

1. Source `SYBASE.csh` or `SYBASE.sh` file if you have not setup the Sybase environment variables.
2. At the UNIX prompt, start `auditinit` (the auditinit utility is located in `$SYBASE/$SYBASE_ASE/install`):

   ```
   $SYBASE/$SYBASE_ASE/install/auditinit
   ```

   `auditinit` displays the following menu:

   ```
   AUDITINIT
   1. Release directory:  /usr/u/sybase
   2. Configure a Server product
   ```

3. Select **Configure a Server Product**.
4. Select **Adaptive Server**.
5. Select **Configure an Existing Sybase Server**.
6. Select the server to configure.
7. Provide the SA password for the server you selected.
8. From the **Sybase Server Configuration** screen, select **Configure Auditing**.

   As you proceed through the menus in `auditinit`, you can change any default values that appear. As you finish each menu, press Ctrl-A to accept the defaults or changed values and move to the next menu.

   ```
   CONFIGURE AUDITING
   1. Configure auditing:  no
   2. Add a device for audit table(s)
   3. Add a device for the audit database transaction log
   4. Delete a device entry
   5. Change a device entry
   List of devices for the audit tables:
   Logical name    Physical name    Segment name    Table name    Size
   Device for the audit database transaction log:
   Logical name    Physical name    Segment name    Table name    Size
   ```

9. From the **Configure Auditing** screen, select **Configure Auditing**.

`auditinit` redisplays the **Configure Auditing** menu with the value "yes" displayed for **Configure Auditing**.

10. Restart the SAP ASE Cluster Edition for the changes to take effect.

40.4.2 Creating a Device for an Audit Table

Create a device or an audit table. To add multiple audit devices, repeat all the steps. You can add as many as eight devices. We recommend adding three or more audit table devices.

Procedure

1. From the **Configure Auditing** screen, select **Add a Device for Audit Table(s)**.

   `auditinit` displays the following menu:

   ```
   ADD/CHANGE A NEW DEVICE FOR AUDITING
   1. sybsecurity physical device name:
   2. Logical name of the device:
   3. Size of the device (Meg):
   4. Device size for auditing:
   ```

2. Select **Sybsecurity Physical Device Name**.
3. Enter the full path of the physical device (raw partition) that you located in preinstallation tasks for auditing devices, where `<path_to_partition>` is the path to the raw partition for the device:

   ```
   Enter the physical name of the device to use for the audit database (default
   is " "):
   /dev/<path_to_partition>
   ```

   If you specify an operating system file, the following warning appears:

   ```
   WARNING: '/secret1/sybase_dr/install/aud1.dat' is a
   regular file which is not recommended for a Server
   device.
   ```

4. Press **Return** to acknowledge the warning. `auditinit` redisplays the **Add/Change a New Device for Auditing** menu, which displays the physical name of the device:

   ```
   ADD/CHANGE A NEW DEVICE FOR AUDITING
   1. sybsecurity physical device
   name:  /secret1/sybase_dr/install/aud1.dat
   2. Logical name of the device:
   3. Size of the device:
   4. Device size for auditing:
   ```

   Note

   The **Size of the Device** value must be equal to or greater than the **Device Size for Auditing** value. The **Device Size for Auditing** must be equal to the device size. If you are following the auditing guidelines, you need not to change the value displayed in **Device Size for Auditing**.

5. Proceed through the remaining items on this menu.
6. Press Ctrl-A to accept the settings. `auditinit` returns to the **Configure Auditing** menu and displays the device you have created:

```
CONFIGURE AUDITING
 1.  Configure auditing:  yes
 2.  Add a device for audit table(s)
 3.  Add a device for the audit database transaction log
 4.  Delete a device entry
 5.  Change a device entry
List of devices for the audit tables:
Logical name   Physical name    Segment name         Table name    Size
6.Audit_01' secret1/sybase_dr/install/aud1.dat' sysaudits_01 5
```

After adding a device, `auditinit` returns to the **Configure Auditing** menu and displays all the devices you have created:

```
CONFIGURE AUDITING
1. Configure auditing:  yes
2. Add a device for audit table(s)
3. Add a device for the audit database transaction log
4. Delete a device entry
5. Change a device entry
List of devices for the audit tables:
Logical name    Physical name    Segment name         Table name     Size
6. Audit_01'   /secret1/sybase_dr/install/aud1.dat’ sysaudits_01   5
7. Audit_02'   /secret1/sybase_dr/install/aud2.dat' sysaudits_02   5
```

Related Information

Preinstallation Tasks for Auditing Devices [page 290]

40.4.3 Creating a Device for the Audit Database Transaction Log

Create a device for the audit database transaction log.

Procedure

1. From the **Configure Auditing** menu, select **Add a Device for the Audit Database Transaction Log**. `auditinit` displays the **Add/Change a New Device for Auditing** menu:

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
1.  sybsecurity physical device name:
2.  Logical name of the device:
3.  Size of the new device (Meg):
4.  Device size for auditing:
```

2. Select **Sybsecurity Physical Device Name**. `auditinit` prompts for the physical name and supplies you with a default, if available:

```
Enter the physical name of the device to use for the
sybsecurity database (default is''):
 /dev/<path_to_partition>
```

   `<path_to_partition>` is the path to the raw partition for the device.

3. Enter the full path name of a physical device. If you enter an operating system file name, the following warning appears:

```
WARNING: '/secret1/sybase_dr/install/audlog' is a regular file, which is not
recommended for a Server device.
```

4. Press **Return** to acknowledge this warning. `auditinit` displays the **Add/Change a New Device for Auditing** menu and the value you selected for the physical name of the device:

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
1.sybsecurity physical device name:
      /secret1/sybase_dr/install/auditlog.dat
2.Logical name of the device:
3.Size of the device:
4.Device size for auditing:
```

5. Proceed through the remaining items on this menu. Note the following:
   - The minimum size for the transaction log is 2 MB.
   - `auditinit` displays the size in both **Size of the Device** and in **Device Size for Auditing** in the **Add/Change a New Device for Auditing** menu.
   - The **Device Size for Auditing** default value is equal to the size of the device, based on the assumption that you may want to devote the entire device to log for the auditing task. If you only want to use a subset of the device, edit the **Size of the Device** value.
6. Press Ctrl-A to accept the settings displayed in the **Add/Change a New Device for Auditing** menu. `auditinit` returns to the **Configure Auditing** menu and displays all the devices you created:

```
CONFIGURE AUDITING
1. Configure auditing:  yes
2. Add a device for audit table(s)
3. Add a device for the audit database transaction log
4. Delete a device entry
5. Change a device entry
List of devices for the audit tables:
Logical name    Physical name    Segment name         Table
name     Size
6. Audit_01'   /secret1/sybase_  dr/install/aud1.dat' sysaudits_01   5
7. Audit_02'   /secret1/sybase_  dr/install/aud2.dat' sysaudits_02   5
8. auditlog    /secret1/.../auditlog.dat logsegment   syslogs        2
```

7. When you are ready to execute the audit configuration, press Ctrl-A. `auditinit` returns you to the **Sybase Server Configuration** screen.
8. Press Ctrl-A again. `auditinit` prompts with:

```
Execute the Sybase Server Configuration now?
```

9. Enter "y" (yes).

`auditinit` executes the tasks to install auditing. When the installation completes successfully, you see the following messages:

```
Running task: install auditing capabilities.
....................Done
Auditing capability installed.
Task succeeded: install auditing capabilities.
Configuration completed successfully.
Press <return> to continue.
```

40.4.4 Enabling Auditing

After auditing is installed, no auditing occurs until a System Security Officer enables auditing with `sp_configure`.

See the *SAP ASE System Administration Guide: Volume 1*.

40.4.4.1 Deleting a Device Entry

Delete a device entry.

Procedure

1. From the **Configure Auditing** menu, select **Delete a Device Entry**.
2. Enter the number of the device to delete.
3. Press **Return**.

40.4.4.2 Changing a Device Entry

Change a device entry.

Procedure

1. From the **Configure Auditing** menu, select **Change a Device Entry**.
2. Enter the number of the device to change. `auditinit` displays the **Add/Change a New Device for Auditing** menu with information on the device you selected:

```
ADD/CHANGE A NEW DEVICE FOR AUDITING
1. sybsecurity physical device name:
```

```
      /secret1/sybase_dr/install/audlog
2. Logical name of the device: aud.log
3. size of the new device (Meg): 5
4. Device size for auditing:5
```

3. Select each remaining entry you want to change.
4. Press Ctrl-A to save the new entries.

41 Install Online Help for Transact-SQL Syntax

The `$SYBASE/$SYBASE_ASE/scripts` directory contains scripts for installing `sybsyntax`, the syntax help database.

You can retrieve this data using `sp_syntax`. See the *Reference Manual: Procedures*.

The `scripts` directory contains one or more of the `sp_syntax` scripts shown in, depending on which SAP products are included with your server:

Table 16: sp_syntax Installation Scripts

Script	Product
`ins_syn_cl`	Open Client Client-Library
`ins_syn_esql`	Embedded SQL
`ins_syn_os`	Open Server
`inssynsq`	Transact-SQL

All SAP ASE installations include the `inssynsq` script, which includes syntax information for Transact-SQL, the system procedures, and the SAP utilities. Execute this script to install the SQL portion of the `sybsyntax` database.

You can install any of these scripts, depending on the need for SAP information on your server. The first script you execute creates the `sybsyntax` database and the required tables and indexes. Any scripts that you execute after the first one add to the existing information in the database. If you reexecute a script, the previously installed rows of information are deleted from the table in the database then reinstalled.

> ⚠ Caution
>
> The `ins_syn_cl` and `ins_syn_os` scripts conflict. Executing both scripts causes errors.

41.1 Default Device for the sybsyntax Database

By default, the `sybsyntax` installation scripts install the `sybsyntax` database on the device that is designated as the default database device.

If you have not used `sp_diskdefault` to change the status of the master device (which is installed as the default disk) or to specify another default device, the scripts install `sybsyntax` on the master device. You should not use this configuration because `sybsyntax` uses valuable space, which is best left available for future expansion of the master database.

To avoid installing `sybsyntax` on the master device, either:

- Use `sp_diskdefault` to specify a default device other than the master device. For information about `sp_diskdefault`.
- Modify each `sybsyntax` installation script that you plan to execute to specify a different device.

See the *Reference Manual: Procedures*.

41.2 Installing sybsyntax

The `sybsyntax` installation script installs the database and the necessary tables and indexes.

Procedure

1. Determine the type - raw partition, logical volume, operating system file, and so on - and location of the device where you plan to store the `sybsyntax` database.
2. Make a copy of the original script. Be sure you can access this copy, in case you experience problems with the edited script.
3. If necessary, use a text editor to edit the script to change the default device from the master device to the device created in step 1.

   Comment out this section, which specifies the default device:

```
/* create the database, if it does not exist */
 if not exists (select name from sysdatabases
 where name = "sybsyntax")
 begin
     /* create the sybsyntax table if it doesn't exist */
     /* is the space left on the default database
     devices > size of model? */
     if (select sum (high-low +1) from sysdevices where status
     & 1 = 1) - (select sum(size) from sysusages, sysdevices
         where vstart >= sysdevices.low
         and vstart <= sysdevices.high
         and sysdevices.status &1 = 1) >
         (select sum(sysusages.size) from sysusages
         where dbid = 3)
     begin
         create database sybsyntax
     end
     else
     begin
         print "There is not enough room on the default
         devices to create the sybsyntax database."
     return
     end
 end
```

4. After you have commented out this entire section, add a line such as the following, where `<device_name>` is the name of the device where you want to install sybsyntax:

```
create database sybsyntax on <device_name>
```

5. Execute the script:

```
isql -U<sa> -P<password> -S<servername> < $SYBASE/$SYBASE_ASE/scripts/inssynsq
```

   where <sa> is the user ID of the system administrator, <password> is the system administrator's password, and <servername> is the SAP ASE where you plan to install the database.

   If you set the DSQUERY environment variable to <servername>, you can replace the server name with $DSQUERY.

6. To ensure that you have installed the sybsyntax database and that it is working correctly, use isql to log in to the server on which you installed the database, and execute sp_syntax:

```
isql -Usa -P<password> -S<servername>
1> sp_syntax "select"
2> go
```

   SAP ASE displays a list of commands that contain the word or word fragment "select."

Important Disclaimers and Legal Information

Coding Samples

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended to better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, unless damages were caused by SAP intentionally or by SAP's gross negligence.

Accessibility

The information contained in the SAP documentation represents SAP's current view of accessibility criteria as of the date of publication; it is in no way intended to be a binding guideline on how to ensure accessibility of software products. SAP in particular disclaims any liability in relation to this document. This disclaimer, however, does not apply in cases of willful misconduct or gross negligence of SAP. Furthermore, this document does not result in any direct or indirect contractual obligations of SAP.

Gender-Neutral Language

As far as possible, SAP documentation is gender neutral. Depending on the context, the reader is addressed directly with "you", or a gender-neutral noun (such as "sales person" or "working days") is used. If when referring to members of both sexes, however, the third-person singular cannot be avoided or a gender-neutral noun does not exist, SAP reserves the right to use the masculine form of the noun and pronoun. This is to ensure that the documentation remains comprehensible.

Internet Hyperlinks

The SAP documentation may contain hyperlinks to the Internet. These hyperlinks are intended to serve as a hint about where to find related information. SAP does not warrant the availability and correctness of this related information or the ability of this information to serve a particular purpose. SAP shall not be liable for any damages caused by the use of related information unless damages have been caused by SAP's gross negligence or willful misconduct. All links are categorized for transparency (see: http://help.sap.com/disclaimer).

go.sap.com/registration/contact.html

© 2016 SAP SE or an SAP affiliate company. All rights reserved.
No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.
Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.
These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.
SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.
Please see http://www.sap.com/corporate-en/legal/copyright/index.epx for additional trademark information and notices.