W. W. Hansen Experimental Physics Laboratory
STANFORD UNIVERSITY
STANFORD, CALIFORNIA 94305 - 4085
Gravity Probe B Relativity Mission

Database Recovery Procedures for Science Network Databases

GP-B Procedure
P1071 Revision - A

August 9, 2005

Approvals

NAME	SIGNATURE	DATE
Carin Kahn Author		
Paul Shestopl *Data Processing IPT*		
Ron Sharbaugh *S/W Manager*		
Marcie Smith *MOC Project Manager*		
Dorrene Ross *Software Quality Assurance*		

Required Signatures prior to Execution

NAME	SIGNATURE	DATE
NAME: *Data Base Administrator*		
Name: System Administrator		
Dorrene Ross *Software Quality Assurance*		

Tom Langenstein ITAR Assessment Performed, ITAR Control Req'd? □ Yes □ No

Table of Contents

1 Revision History 2
2 Scope 3
3 Reference Documents 3
4 Operational Personnel 3
5 QA Provisions 4
6 Recovery Scenario Start 4
7 Recovering User Databases 5
7.1 Time estimates for recovery of user databases 5
7.2 Overview of backup architecture 6
7.3 Recovering a user database 8
7.4 Worst-Case Scenario: Making the last 48 hours worth of data available for Level 0, 1 and 2 databases 15
8 Recovering System Databases 17
8.1 Recovering the master Database or Device 17
8.2 Recovering the model Database 32
8.3 Recovering the sybsystemprocs Database 34
8.4 Recovering the sybsystemdb Database 36
8.5 Recovering the sybsyntax Database 38
8.6 Recovering the dbccdb Database 39
8.7 Recovering the sybdba Database 40
9 Miscellaneous Tasks 41
9.1 Dropping & Recreating a Damaged Database 41
9.2 Restoring a file from ufsbackup on tape (OS backups run by SA) 42

1 Revision History

REV	DATE	AUTHOR	COMMENTS
-	2/1/04	CWK	Initial version
A	8/9/05	CWK	Adjust to installation of new hardware

2 Scope

In the event of a database failure or a system failure that may require database recovery it is imperative that the database administrator or designee is contacted immediately. The dba/designee will help identify the problem and assess what recovery strategy to employ.

The purpose of this document is to provide step-by-step instructions for the various Sybase database recovery scenarios that involve loading one or several databases from backup.

When a fairly large database has to be recovered or several databases are affected, the database outage may be significantly shortened by switching over to the standby server before initiating the recovery process. It is the responsibility of the dba to investigate the different recovery paths and present to the arb for a decision. The recovery review process, as well as the procedure for switching over to the standby server are addressed in documents P1081 and P1070, “Assessment of Science Network Database Errors and Failures” and “Database Recovery Procedures Using the Standby Database Server”

This document does not address the routine preloading of backups to the warm standby server that are done to minimize downtime during an unscheduled switchover. This is covered in document P0996, “Database Backup Procedures for Science Network Databases”

The science_prod and science_pc database servers are running version 12.5.0.3 as of October 2004. It is important to understand however that this document does not apply for database servers running a version lower than 12.5.x. The section on system database recovery may also not work for servers that were originally created in a version lower than 12.5 and then upgraded to 12.5.x. If a pre-12.5 Sybase server needs to be upgraded, the server should be created from scratch as a 12.5.x server and the data ported using the bulk copy facility.

3 Reference Documents

Document	Document No.	ALIAS
Data Management Plan	S0331	
Science Data Network 24 Hour Recovery Plan During IOC	S0976	
Database Backup and Recovery Plan for Science Network Databases	S0979	
Database Backup Procedures for Science Network Databases	P0996	
Database Recovery Procedures Using the Standby Database Server	P1070	
Assessment of Science Network Database Errors and Failures	P1081	
Telemetry Data Processing (TDP) in the Non-Real-Time System	P0826	

4 Operational Personnel

This procedure may only be conducted by the following persons:

- Carin Kahn or the designated Database Administrator
- Dorrene Ross Qualified QA Rep

5 QA Provisions

This procedure shall be conducted on a formal basis to its latest approved and released version. Software QA (Dorrene Ross) shall be notified 24 hours prior to the start of this procedure. QA may monitor the execution of all or part of this procedure should they elect to do so.

In case of an emergency it will not be possible to adhere to the 24 hour notification-rule. If the Data Processing IPT gives the go-ahead to implement a procedure, QA is notified by pager and by phone.

QA notification time/date:

Date/time:__________________

GP-B QA (Dorrene Ross)

6 Recovery Scenario Start

Notification of mission/flight director has been made prior to start of recovery.

Recovery Scenario(s) Used:

____ started __________ ended _________

____ started __________ ended _________

____ started __________ ended _________

____ started __________ ended _________

Executed by _________________________________ Signature: _______________________________

Witnessed by ________________________________ Signature: _______________________________

7 Recovering User Databases

A user databases is a database that is not a system database. User databases contain user data and their setup and contents differ from installation to installation. System databases control the user databases and the Sybase dbms itself; they are typically very similar from installation to installation. For the purposes of this document the user database sybdba, which contains space usage data is treated as a system database.

To ensure continuous backup and recovery, full backups are made weekly and the transaction logs backed up every 30 minutes. Recovery of a database always starts with the loading of a full backup. After the full backup has been loaded the transaction logs are applied up to the desired point in time. It is possible to apply only parts of a transaction log file.

7.1 Time estimates for recovery of user databases

The recovery-scripts written for Gravity Probe B all have a feature where the name of a dump-file that has been loaded to a database is added to a load-log. The dump-file names added to this load-log include the date and time that the backups were initially created, making it easy to quickly determine how long it would take to bring a standby database up to date or to simply monitor loading progress during recovery. The load-logs are located in /home/sybase/sybsystem/<server>/load_log.<server>.<database>.

The following time estimates are just that, estimates. The actual recovery time will depend on what other activity is taking place on the affected database server etc. This data is provided as a guideline only.

Recovering full backups	Total size of database
Recover a full backup from disk to science_pc	70-90 gb/hr
Recover a full backup from disk to science_prod	50-80 gb/hr
Recover a full backup from local tape drive	50-80 gb/hr
Recover a full backup from remote tape drive	5-15 gb/hr

An effort is made by the dba to run the full backups when there is minimal update activity on the database. That is not always possible however and the recovery time may be extended by up to as much as 70% if the database was very active during the backup. This additional recovery time is needed for applying the changes in the transaction log to the database and checkpointing while bringing the database online.

Recovering transaction logs	Size of tranlog dumps	
Recover transaction logs from disk	3-6 gb/hr	
Recover one day's worth of logs for database	GPB_L0	5-16 minutes
	GPB_L1	8-10 minutes
	GPB_L1A	70-100 minutes
	L2	5-14 minutes
	Orbit_determ	1-4 minutes
	MSS-database	1-2 minutes
	Total Time	90-146 minutes
	Common Time	80-100 minutes

7.2 Overview of backup architecture

The full backups of user databases are initially written to a backup disk attached to the sci-crunch server. After being preloaded to the standby server they are subsequently archived to dual tapes. The backups on tape are retained for three months with one set of backups per month kept for the life of the mission.

Due to time constraints a large enough backup disk could not be attached to sci-crunch before start of IOC. Two months after IOC the full backups thus no longer fit on the backup disk and they were instead made directly to tape from the science_prod databases. (Larger backup disks were configured and attached to sci-crunch after IOC).

Transaction logs are backed up to a backup disk attached to the sci-base host and also copied to the Sybase home directory several times a day. The Sybase home directory is backed up to tape weeknights as part of regular backup system maintained by the SA. The transaction logs are retained for four weeks on the sci-base backup disk and one week in the Sybase home directory.

The tape drives on the science-base and science-crunch servers have been configured the exact same way. A tape written on one tape drive can thus be read on the other and vice versa. The configuration must not be altered, a change would render existing tapes unreadable. Although the tape drives are interchangeable a remote tape dump or load may take five to ten times as long as the corresponding local action. To verify the setup issue:

- `mt –f /dev/rmt/0cn status`

Note that two slightly different outputs are valid:

MIRROR with two tapes loaded or AUTOLOAD with one tape loaded:

```
Sony AIT-x 8mm tape drive:
 sense key(0x6)= Unit Attention   residual= 0   retries= 0 file no= 0   block no= 0
```

MIRROR with one tape loaded or AUTOLOAD with two tapes loaded:

```
Sony AIT-x 8mm tape drive:
 sense key(0x0)= No additional sensor   residual= 0   retries= 0 file no= 0   block no= 0
```

- `cat /kernel/drv/st.conf`

locate a tape-config-list label with the following setup (others may co-exist):

```
tape-config-list=
"SONY    SD", "Sony AIT-x 8mm", "SONY_AIT"
SONY_AIT = 1, 0x36, 0, 0xd679, 4, 0x00, 0x00, 0x00, 0x00, 0
```

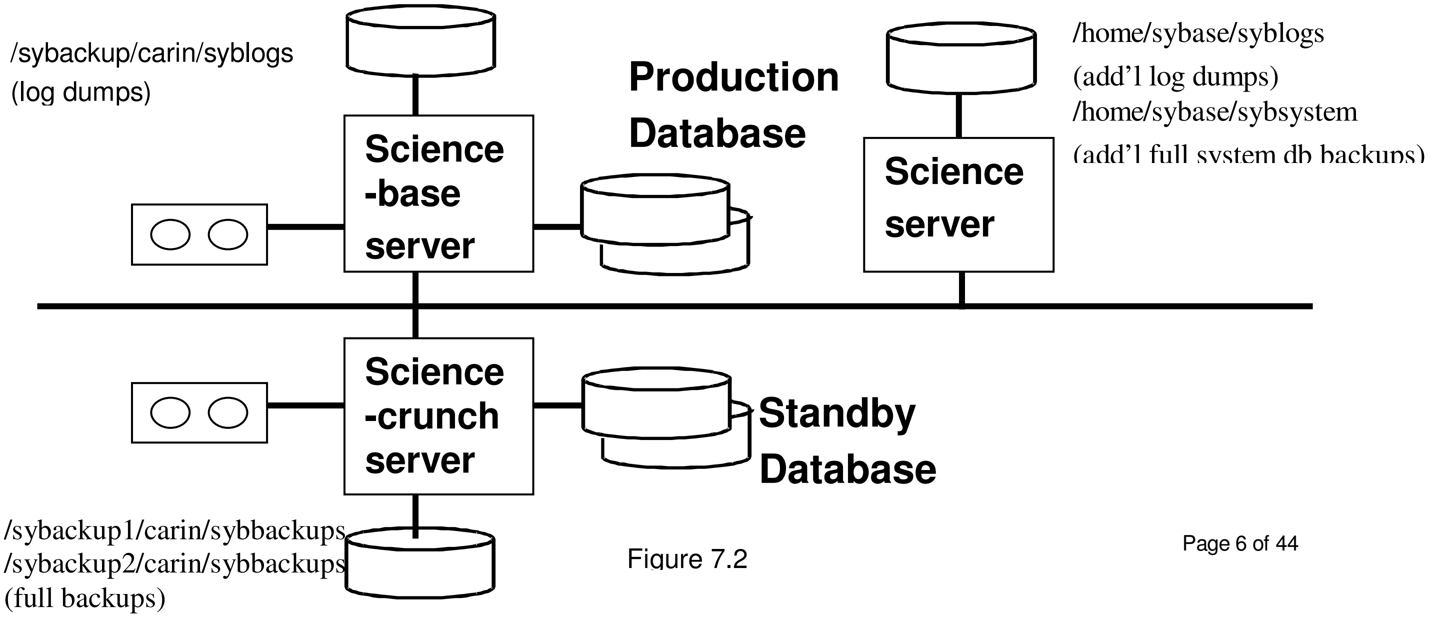

Figure 7.2

7.3 Recovering a user database

Start by recovering the full database from either disk or tape. Load the transaction logs up to the desired point in time after the full database recovery has completed successfully.

7.3.1 Recover the full database from the disk

7.3.1.1 If you are recovering a production database in place, that is, you will be overwriting the existing database, dump the transaction log if possible before initiating the load. This will ensure up-to-the-minute recovery in case you'll need to revert back to the version you're overwriting

```
~/sybdba/script/tranlog_backup.csh <server> <database>
```

____

7.3.1.2 Load the database from disk. **Do not online the database after the load completes**. Putting the database online makes an entry in the transaction log, preventing subsequent tranlogs from loading. All eight parameters are mandatory.

```
cd /home/sybase/sybdba/script
recover_db.csh <original_server> <server_to_load_to>  \
      <original_database(s)_to_load> <database_to_load_to_or_"same">  \
      <generation_to_load_from> <dbcc/nodbcc switch>
      <prefix_for_log_file> "<Informational text for log file>"
```

- to reload the most recent GPB_L0 db on science_prod

```
recover_db.csh science_prod science_prod GPB_L0 same gen_0 nodbcc reload "Reload"
```

- to reload a previous version of MSS databases

```
recover_db.csh science_prod science_prod gpb_3_4_2_1.gpb_3_4_2 same gen_1 dbcc reload "Reload"
```

- to reload GPB_L1A under a different name

```
recover_db.csh science_prod science_prod GPB_L1A CWK_L1A gen_0 nodbcc copy "Copy"
```

- scripts for common db recoveries exist, for instance to load databases from production to the standby server or from the standby server to the offlined production server respectively: preload_standby.csh or switchback_offline_db.csh
- Document the command used and execute it:

____ _______

7.3.1.3 The database should have been put online for standby access by the loading script in the previous step. If you need to review the contents of the database and that did not happen, put it online for standby access at this point. Failure to use the standby_access option bars successive transaction logs from being loaded, you'd have to start over with loading the full database backup

```
isql -U<user> -P<password> -S<server>
online database <db> for standby_access
```

____

7.3.1.4 Load the transaction logs per the instructions in section 7.3.3 below, "Recovering the transaction logs of a database" _______

7.3.1.5 Are you absolutely sure that all transaction logs have been loaded and this is the database to make available to users for update? If so, put it online

```
isql -U<user> -P<password> -S<server>
online database <db>
```

____

7.3.1.6 Make a full backup of the newly loaded database. This must be done before resuming tranlog dumps

```
~/sybdba/script/full_backup.csh <server> <db>
OR
Mount a tape on the host where the database server is running
ssh -l sybase <host where the database server is running>
/home/sybase/sybdba/script/tape_backup.csh  \
  <server> <YYMMDD> <backupserver> <db(s)> <init/noinit> <unload/nounload>
```

____

7.3.1.7 Verify that tranlog dumps are restarted in cron if they were suspended _______

7.3.2 Recover the database from tape

7.3.2.1 Locate the appropriate tape that contains the database backup you wish to load. One set of tapes can be found in the dba-office, one of the offices off the MOC. The other set of tapes is stored in a box under the sci-crunch server in room 127 in HEPL. Each tape is marked with the date the backups were taken and a dumpvolume number in the format YYMMDD ______

7.3.2.2 Write-protect the tape by flipping the tab on it to the “safe” position. Mount the tape. If possible, perform a local and not a remote load. (I.e. use the tape drive that is attached to the host where you are loading the database). A remote load may take 5 to 10 times longer than a local load. If the dump spans more than one tape, insert the first tape in the left bay, the second tape in the right

____

7.3.2.3 If you wish to verify a dumpvolume or list the files that are located on a particular tape or tape volume, insert the tape into the tape drive and issue the following command

```
isql –Usa –S<server> -w222
load transaction tempdb from "/dev/rmt/0cn" with listonly=full
```

Dumpvolume is stored as “Volume id:”, highlighted below. Note that the leading 0 in the year 04 is dropped since the info is stored as an integer. The file name is always the source database name.

```
Backup Server session id is:  43.  Use this value when executing the
'sp_volchanged' system stored procedure after fulfilling any volume change request
from the Backup Server.
Backup Server: 4.35.1.1: Device '/dev/rmt/0cn':
Label name:     'VOL1'
Volume id:      '40712 '
Access code:    ' '
Reserved:       '                          '
Owner id:       '              '
Reserved:       '                            '
Labeling
version:        7

Backup Server: 4.37.1.1: Device '/dev/rmt/0cn':
Label id:       'HDR1'
File name:      'GPB_L1A          '
Stripe count:   1
Device typecount:       1
Archive volume number:  1
Stripe position:        0
Generation number:      0001
Generation version:
00

Backup Server: 4.148.1.1:
Create date & time:     Wednesday, Jul 14, 2004, 14:35:26
Expiration date & time: Wednesday, Jul 14, 2004, 00:00:00
Access code:    ' '
File block count:       0
Sybase id string:       'Sybase  '
Reserved:       '   '

Etc. . . . .
```

______

7.3.2.4 Determine if you need to change the ASP-mode of the tape drive, it specifies how the two bays are handled when recovering from multiple volumes. The current setting is displayed on the left-hand LCD-display of the tape drive. If you are loading from one tape volume only the LCD-displays should say “Mirror”. If you are loading from several tape volumes the left LCD-display should say “Autoload Idle”, showing it’s the master, and the right “Inactive”. (The idle/inactive settings only displays when a tape is actually loaded in the bay). If the setting is Autoload but the left bay is not set to Idle, switch by first setting the ASP-mode back to Mirror and then resetting it to Autoload (see the next step).

____

7.3.2.5 Change the ASP-mode using the three push-buttons marked Menu, Select and Enter on the right-hand side of the tape drive. Current settings and messages while making changes are displayed in the left-hand LCD-display window. The settings cannot be changed while the tape drive is in use

```
Push Menu to initiate a change
=> Configuration menu
Push Enter to change a configuration
Push Select until “ASP mode” is displayed
Push Enter to change the ASP mode
Push Select until the desired ASP mode is displayed. It is Mirror if
  loading from one tape volume, Autoload if loading from several
Push Enter when the correct ASP mode is displayed
Push Menu to exit
Push Enter to save the changes before exiting; Menu to cancel
Verify the settings are correct                                   _______
```

7.3.2.6 If you are recovering a production database in place, that is, you will be overwriting the existing database, dump the transaction log for the database to be overwritten, if possible, before initiating the load. This will ensure up-to-the-minute recovery in case you’ll need to revert back to the version you’re overwriting

```
~/sybdba/script/tranlog_backup.csh <server> <database>            _______
```

7.3.2.7 Start the load from tape. Because of some tape-handling sequences in the script, the load must be started from the host where the tape drive is attached. **Do not online the database after the load completes**. Putting the database online makes an entry in the transaction log, preventing subsequent tranlogs from loading. The first three parameters are mandatory

```
ssh -l sybase <host where tape drive is attached>
cd /home/sybase/sybdba/script
tape_load <target sybase server> <dumpvolume on tape>
      <backup server running on host where the tape drive is attached>
      <original_database(s)_to_load> <database_to_load_to_or_”same”>  \
      <dbcc/nodbcc switch> <unload/nounload switch>
•     load GPB_L0/L1 from 6/15/04 to science_prod, using tape-drive on sci-
base and unloading the tape when the last file has been loaded
tape_load.csh science_prod 40615 science_prod_back GPB_L0.GPB_L1 same nodbcc
unload
•     remote load from tape drive on sci-crunch to databases on science_prod
tape_load.csh science_prod 040615 science_pc_back gpb_3_4_3 same dbcc nounload
•     Document the command used and execute it:
```

____   _______

7.3.2.8 Does the dump span more than one tape volume? When end-of-tape is reached for the first volume the load is suspended and an entry made in the backup-log indicating that the tape volumes need to be changed. This log-entry initiates a page to the dba with the same msg. The second tape should have been inserted in the right bay of the tape drive in step 7.3.2.2 above. If not, do so now. When the tape has been inserted and queued, log in to the sybase server to let the backup server know that the new tape is available by issuing the sp_volchanged command

```
isql -Usa -S<server> -w222
copy and paste the sp_volchanged command from the backup-log.
For example: exec sp_volchanged
            @session_id = <id from log>,
            @dev_name = = “/dev/rmt/0cn”,
                @action = "PROCEED"
```

____

7.3.2.9 If the dump spans more than two tape volumes, insert the third tape into the left-hand bay once the first tape has been automatically ejected. Issue another sp_volchanged cmd when this tape is requested. If the dump spans more than three tapes, insert the fourth tape into the right-hand bay once the second tape has been automatically ejected …. And so on until the load has completed

____

7.3.2.10 The database should have been put online for standby access by the loading script in the previous step. If you need to review the contents of the database and that did not happen, put it online for standby access at this point. Failure to use the standby_access option bars successive transaction logs from being loaded, you'd have to start over with loading the full database backup

```
isql –U<user> –P<password> –S<server>
online database <db> for standby_access
```

____

7.3.2.11 Load the transaction logs per the instructions in section 7.3.3 below, "Recovering the transaction logs of a database"

____

7.3.2.12 Are you absolutely sure that all transaction logs have been loaded and this is the database to make available to users for update? If so, put it online

```
isql –U<user> –P<password> –S<server>
online database <db>
```

____

7.3.2.13 Make a full backup of the newly loaded database. This is necessary before resuming tranlog dumps

```
~/sybdba/script/full_backup.csh <server> <db>
OR
Mount tape on the host where the database server is running
ssh –l sybase <host where the database server is running>
/home/sybase/sybdba/script/tape_backup.csh I \
  <server> <YYMMDD> <backupserver> <db(s)> <init/noinit> <unload/nounload>
```

____

7.3.2.14 Verify that tranlog dumps are restarted with cron if they were suspended

____

7.3.3 Recovering the transaction logs of a database

The recovery of a transaction log is done in two steps by Sybase. In the first step the log itself is loaded to the server and in the second step the changed pages as recorded in the tranlog are applied to the database. Applying the changes may take several times longer than the actual loading of the log file.

In most cases it is sufficient to simply load a set of transaction logs that were created before a particular point in time. In some cases, for instance if you know **exactly** when a database corruption occurred, you might want to load only the transactions that were committed before a specific point in time, a time that falls within the boundaries of a tranlog. This latter option should be used with caution. Once the last, partial log has been loaded an entry is automatically made in the new tranlog. This log entry prevents the loading of any more tranlogs. If you wish to load additional logs at this point you will have to start all over with loading a full backup.

Option A: Loading a set of transaction logs

7.3.3.1 Determine the lag-time in hours for loading the tranlogs. In other words, what's the cutoff time before which the tranlog dump must have been started in order for it to be loaded.

Lag time:

0	means all tranlogs dumps started before "now" will be loaded
24	means all tranlog dumps started at the latest 24 hours ago will be loaded
1.5	means all tranlog dumps started at least 1.5 hours ago will be loaded

Determine timestamp of the last tranlog dump you wish to load:

```
ls –l /sybackup/carin/syblogs/<server>/<db>/<gen>/
```

> lists files in the format <db>.log.yymmddhhmmss, for instance GPB_L1.log.20040713200523
the last 14 digits are compared to the lag-time and determines if the tranlog is loaded or not

Determine what lag-time to use when executing the tranlog script:

```
isql –Usa –Sscience_pc –P<password>
select datediff( minute, “mm/dd/yy hhmm am/pm”, getdate())/60.  – don’t forget decimal point after 60
-- round off the number you get and verify that the cutoff time is the one you want
select dateadd (minute, -<rounded off number from previous query>*60, getdate())
```

7.3.3.2 Start loading the tranlogs

```
cd /home/sybase/sybdba/script
recover_log.csh <original_server> <server_to_load_to>  \
      <original_database(s)_to_load> <database_to_load_to_or_”same”>  \
      <lag-time in hours as calc above> <dbcc/nodbcc switch>
      <prefix_for_log_file> “<Informational text for log file>”
```

- to reload all the tranlogs for GPB_L0

```
recover_log.csh science_prod science_prod GPB_L0 same 0 nodbcc reload “Reload”
```

- to reload tranlogs up to 5.5 hours ago for level 0 and 1 databases

```
recover_log.csh science_prod science_prod GPB_L0.GPB_L1.GPB_L1A same 5.5 dbcc
reload “Reload”
```

- to reload tranlogs from GPB_L1A to a different database and server

```
recover_log.csh science_prod science_pc GPB_L1A CWK_L1A 24 nodbcc copy “Copy L1A”
```

- scripts for common log recoveries exist, for instance to load logs from production to the standby server or from the standby server to the offlined production server respectively:

```
preload_tranlog.csh or switchback_offline_log.csh
```

- Document the command used and execute it:

____ ______

7.3.3.3 Option B: Loading committed transactions up to a specific point in time (partially loading the last tranlog)
Start loading the tranlogs

```
cd /home/sybase/sybdba/script
recover_log_until.csh <original_server> <server_to_load_to>  \
      <original_database(s)_to_load> <database_to_load_to_or_”same”>  \
      “<load committed trans up to date & time>” <dbcc/nodbcc switch>
      <prefix_for_log_file> “<Informational text for log file>”
```

- Examples:

```
recover_log_until.csh science_prod science_prod GPB_L0 CWK_L0  \
      “7/13/04 5:35pm” nodbcc reload “Reload of science”
recover_log_until.csh science_prod science_prod GPB_L0.GPB_L1.GPB_L1A same \
      “6/18/2004 17:10:05” dbcc reload “Reload”
```

- Document the command used and execute it:

____ ______

7.3.3.4 Complete the load sequence by returning to where you left off, either in step 7.3.1.5 or 7.3.2.12 above, when you are convinced that all the transaction logs have been loaded

____

7.3.4 Contingency: The transaction log cannot be dumped due to device failures

7.3.4.1 If there are device problems, you may not be able to truncate the transaction log. Try dumping the log without truncating it with the command below. If that doesn’t work the you must proceed without rescuing the last transaction log.

```
isql –Usa –P<password> –S<server>
```

```
dump tran <db> to
“/home/sybase/syblogs/<server>/<db>/gen_0/<db>.log.ccyymmddhhmmss”
with no_truncate                                                  _______
```

7.3.5 Contingency: Manual load of the full database backup from disk

7.3.5.1 To manually load the full backup issue the following command

```
isql –U<user> –P<password> –S<server>
load database <database> from
 “/home/carin/sybbackups/<server>/<db>/gen_<gen>/<db>.full.<date>”_______
```

7.3.5.2 Make an entry in the load_log for documentation purposes and so that successive transaction logs may be loaded automatically, triggering on the date of the last loaded file (14 digits)

```
vi /home/sybase/sybsystem/<server>/load_log.<server>.<database>
add for instance
```

/sybackup1/carin/sybbackups/science_prod/GPB_L1/gen_0/GPB_L1.full.20040520153214

- entry to load_log-file named ______________________________

entry:

____ _______

7.3.6 Contingency: Manual load of the full database backup from tape

7.3.6.1 To manually load the full backup from tape issue the following command

```
isql –U<user> –P<password> –S<target server>
-- local tape drive
load database <target db> from “/dev/rmt/0cn”
  with dumpvolume = YYMMDD, file = <source db>
-- or, less useful, remote tape drive
load database <target db> from “/dev/rmt/0cn”  at <remote backup server>
  with dumpvolume = YYMMDD, file = <source db>                   _______
```

7.3.6.2 Make an entry in the load_log for documentation purposes and so that successive transaction logs may be loaded automatically, triggering on the date of the last loaded file (14 digits, use 0’s for the time)

```
vi /home/sybase/sybsystem/<server>/load_log.<server>.<database>
add for instance
```

tape:GPB_L1.full.20040523000000

- entry to load_log-file named ______________________________

entry:

____ _______

7.3.7 Contingency: Manually loading the tranlog dumps

7.3.7.1 To manually load transaction logs issue the following command

```
isql –U<user> –P<password> –S<server>
load transaction <database> from
 “/home/carin/syblogs/<server>/<db>/gen_<gen>/<db>.log.<date>”     _______
```

7.3.7.2 Make an entry in the load_log for every tranlog that was loaded. This is for documentation purposes and so that successive transaction logs may be loaded automatically, triggering on the date of the last loaded file (14 digits)

```
vi /home/sybase/sybsystem/<server>/load_log.<server>.<database>
add for instance
```

/sybackup/carin/syblogs/science_prod/GPB_L1/gen_0/GPB_L1.log.20040523150534

/sybackup/carin/syblogs/science_prod/GPB_L1/gen_0/GPB_L1.log.20040523153531

- entries to load_log-file named ______________________________

entry:

7.3.8 Contingency: The tape drive seems to be hung

Occasionally the tape drive may hang if an active dump or load was interrupted. The following details how to make the tape drive available again.

7.3.8.1 Shut down the backup server. Wait until any ongoing dumps or loads have completed if possible.
isql –Usa –S<sybase server linked to backup server>
shutdown SYB_BACKUP with nowait _______

7.3.8.2 If the backup server is still running, kill the backup server process from the operating system

```
ssh –l sybase <host where backup server is running>
ps –fu sybase | grep backup server
kill <spid>
or if that doesn't work kill -9 <spid>
```

____

7.3.8.3 Make sure there are no lingering dump/load processes

```
ssh –l sybase <host where backup server is running>
ps –fu sybase | grep multbuf
pf –fu sybase | grep –E "load|dump|recover|backup"
if any processes are displayed, kill <spid>
```

____

7.3.8.4 Restart the backup server
ssh –l sybase <host where backup server is running>

```
cd $SYBASE/ASE/install
./startserver –f RUN_<server>_back
```

____

7.3.8.5 Is the tape drive still busy although it is not being used by any dump/load job?
ssh –l sybase <host where backup server is running>
mt –f /dev/rmt/0cn status => device busy _______

7.3.8.6 Reboot the tape drive itself by powering off and on using the button on the back of the drive_____

7.3.9 Contingency: Recover tranlog dumps from backup directory before loading

If the tranlogs you need to load from the /sybackup/carin/syblogs backup disk cannot be used, you should be able to find them for the last week in the Sybase home directory. You must copy these files to the /sybackup/carin/syblogs directory before they can be loaded using the existing load scripts.

7.3.9.1 Copy the files to the sybase home directory

```
ssh –l sybase sci5
cd /home/sybase/syblogs/<server>/<db>/<gen>
cp /export/home1/syblogs/<server>/<db>/<gen>/<db>.log.<date-criteria> .
```

____

7.3.9.2 Continue loading the tranlog dumps with as planned _______

7.3.10 Contingency: Recover tranlog dumps from OS backup tape before loading

If the tranlog dumps you wish to restore cannot be found in either the /sybackup1/carin/syblogs directory or the Sybase home directory, you should be able to recover them for the last month from an OS backup tape. Incremental backup tapes are generated Monday thru Thursday and recycled weekly. Weekly incrementals are generated every Saturday and recycled monthly. Full backups are generated monthly and never recycled. This means that you should be able to recover a tranlog dump from one of these tapes for the last month. The log dump files must be restored from tape to disk before they can be loaded using the regular Sybase scripts.

7.3.10.1 Restore the transaction log dumps from the OS backup tapes according to the procedure documented in section 9.2 below, "Restoring a file from ufsbackup on tape (OS backups run by SA)" ____

7.3.10.2 Continue loading the tranlog dumps as planned _______

7.4 Worst-Case Scenario: Making the last 48 hours worth of data available for Level 0, 1 and 2 databases

As the databases grow full recovery will take longer and longer. In the unlikely event that it is not possible to fully restore a failing database within the allotted 24-hour window and it is also not an option to switch over to the standby server and bring it up to date, the last resort is to make the last 48 hours worth of data available immediately and work on the long-term recovery option in parallel.

This disaster scenario is very unlikely. The path to take and the details of the implementation will depend on the specific circumstances and set of events that preceded the database failure. Evaluate the various recovery alternatives outlined below and chose the most appropriate one to customize and implement. The recovery procedures are provided as guidelines only, they have not been tested in detail.

7.4.1 Recover an out-of-date backup, process 48 hours and backfill later

This assumes that you will be able to load a backup in place, overwriting the existing database. Having to drop and recreate the database before loading a backup will add considerably to the recovery time.

7.4.1.1 Recover the last viable backup and possibly transaction logs for the database

7.4.1.2 Bring the database online.

7.4.1.3 Populate the database with the last 48 hours worth of data. This is done by the data processing team for level 0 and 1 databases and by the science team for the level 2 database.

7.4.1.4 Notify the end users that the most recent data is now available

7.4.1.5 Backfill the gap in the data, i.e. add data that was not recovered and that is older than 48 hours. The data processing or science team is responsible for this reprocessing of data as well. Additional data that’s coming in is processed in parallel to the reprocessing

7.4.1.6 Notify the end users when all data is available, reprocessing has caught up

7.4.2 Add 48 hours of data to an empty database, recreate a complete db in parallel

If it is crucial to give access to the last 48 hours of data in next to no time, it is probably faster to create a new, empty database to hold only a few days worth of data for immediate use. Once the smaller database has been populated, proceed with the complete recovery of the database in its entirety in a different place and under a different name. Once the recovered database is up to date it is renamed and switched in for the smaller database.

7.4.2.1 Make the failing database unavailable to the end users. You may drop it or simply hide it from the end users by changing its name to FAILED_<db> and then setting it to dbo use only. Since there are no stored procedures or views that reference the level 0, 1, and 2 databases by name/database-id there should not be any problems with applications accessing the wrong database by mistake.

7.4.2.2 Create a new, smaller database to contain the last 48 hours worth of data. This database has the proper name, GPB_L0, GPB_L1, GPB_L1A or L2.

7.4.2.3 Add tables and other objects that need to reside in the new, smaller database

7.4.2.4 Populate the new database with the last 48 hours worth of data. This is done by the data processing team for level 0 and 1 databases, thru reprocessing data files, and by the science team for the level 2 database.

7.4.2.5 Recreate a full-sized version of the failing database if necessary. Call it FAILED_<db>. This can either be made in place on the primary server or on the standby server.

7.4.2.6 Load the latest viable backup and possibly transaction logs to FAILED-<db>.

7.4.2.7 Bring the FAILED-<db> online

7.4.2.8 Reprocess all the missing data, i.e. data that was added after the date of recovery. This is done by the data processing or science team as appropriate.

7.4.2.9 If the reprocessing was done on the standby server, copy the database from the standby to the primary server once the database has been brought up to date. Use the regular backup and recovery scripts.

7.4.2.10 Substitute the fully recovered/reprocessed database for the smaller database. This is done by switching the names of the databases. The smaller database is renamed to NEW_<db> and the recovered database gets the target name, GPB_L0, GPB_L1 etc.

8 Recovering System Databases

A system databases is a database that is not a user database. User databases contain user data and the setup and contents differ from installation to installation. System databases control the user databases and the Sybase dbms itself; they are very similar from installation to installation. For the purposes of this document the user database sybdba, which contains space usage data, is treated as a system database.

Full backups of the master database are run on a daily basis. In addition, important tables are copied out and scripts for recreating disk devices and databases generated daily. Ten generations of these files are all stored in the /sybackup1/carin/sybbackups directory. In addition, three most recent generations ar copied to the Sybase home directory on a daily basis. The Sybase home directory is backed up to tape nightly and weekly by the sa, as part of the regular disk backup system.

Full backups of system databases other than master are made by the dba on an as-needed basis. The backups are made to the /home/sybase/sybsystem/<server> directory. This directory is backed up to tape by the sa weeknights, as part of the regular disk backup system.

8.1 Recovering the master Database or Device

The master database is truly the master of all databases in a Sybase dbms. Tables in this database control all Sybase functions, databases and devices. The Sybase server cannot start or run without a properly functioning master database. Because of its unique role, recovery of a master database is very different from all other databases, regular recovery procedures are **not** used.

Do not initiate recovery of the master database or device unless you are certain that it is the correct course of action. Do not load a backup taken from a different Sybase instance or Sybase version. Do not load a backup that is out of date. As an example, loading incorrect sysusages or sysdevices tables might cause irreversible corruption to the user databases, necessitating a very time-consuming recovery of them as well, once master database functionality has been restored.

If a particular script, file or backup file cannot be found in the /sybackup1/carin/sybbackups/<server>/master directory, refer to the contingencies at the end of this section for procedures to recover files from the Sybase home directory or from an OS backup tape.

8.1.1 Loading a backup of the master database when Sybase is functional

The following process may be used when the Sybase server is still functional but you need to load a backup of the master database for some reason. Do not do this lightly however, the result may be disastrous if not done correctly.

Estimated time to implement: 15-20 min + time for dbcc’s on user databases as deemed necessary

8.1.1.1 Disable any crontab jobs that might try to log in to this Sybase server _______

8.1.1.2 Prevent users from logging in to the sybase server by locking all logins except “sa”, “carin” and “probe”. Log in to the sybase server as a system administrator and issue the following:

```
isql –U<user> –S<server>
select ‘exec sp_locklogin ‘ + name +’, ‘”lock”’
      from master..syslogins where name not in (“sa”, ”carin”, “probe”)
```

execute the generated sql commands

```
select name from master..syslogins where status & 2 != 2
```

The only logins returned should be carin, sa and probe _______

8.1.1.3 Dump the transaction logs in the user databases to guard against possible data loss if master recovery fails. (Do **not** dump the logs if this is the standby server though)

```
/home/sybase/sybdba/script/tranlog_backup.csh <server>                 _______
```

8.1.1.4 Generate up-to-date printouts of the various system tables. This will show you exactly what the master device looked like before the rebuild.

```
/home/sybase/sybdba/script/save_master_info.csh <server>               _______
```

8.1.1.5 Verify you have a valid configuration file, <server>.cfg, in the $SYBASE/ASE-12_5 directory. The number of devices-configuration parameter must be set sufficiently high to avoid losing user databases. The highest device number used can be found in the latest errorlog. Add 1 to account for device number 0 if you have to manually adjust this parameter in the configuration file. _______

8.1.1.6 Shut down the Sybase server nicely to ensure database checkpointing.

```
ssh –l sybase <host where Sybase is running>
/apps/licensed/sybase_local/etc/sybase_stop                            _______
```

8.1.1.7 Make an extra OS-backup of the master device so you can start all over if the master recovery fails. The master device is the –d parameter in the server's RUN_server-file. The backup uses 1GB disk space. To restore, reverse the input and output files - if /of

```
ssh –l sybase <host where Sybase is running>
dd if=<device name>  of=<filename> bs=1024k
example: dd if=/dev/vx/rdsk/vol01_01 of=/home/sybase/mstr.sci.dd bs=1024k    _______
```

8.1.1.8 Start the Sybase server in single-user mode

```
ssh –l sybase <host for sybase server>    # sci-base/sci-crunch
cd $SYBASE/ASE-12_5/install
```

cp RUN_<server> RUN_<server>_single

append –m at end of the file

./startserver –f RUN_<server>_single _______

8.1.1.9 Load the master backup from disk. If the desired backup file is no longer present in the /sybackup1/carin/sybbackups/<server>/master/<gen> directory, it may still reside in the /home/sybase/sybsystem/<server>/master/<gen> directory. If that is the case, copy the file to the /sybackup1/carin/sybbackups directory to make it accessible to the load script. If the backup file cannot be found in either directory, load it from an os-system backup tape, see section 9.2 below, "Restoring a file from ufsbackup on tape (OS backups run by SA)". The Sybase server shuts down after the load is complete

```
isql –Usa –P<password> –S<server>
!!ls –l "/home/sybase/sybsystem/<server>/master/gen_<gen>/master.full.*
load database master from
"/home/sybase/sybsystem/<server>/master/gen_<gen>/master.full.<date>"   _______
```

8.1.1.10 Check for error messages in the output from the isql-session and in the errorlog $SYBASE/ASE-12_5/install/<server>.log. You can safely ignore error msg 930, failing to open the master database, as long as there is a msg stating that the master database is online further down. The following sample output signifies a successful master db load:

```
1> load database master from
"/carin/sybbackups/science_pc/master/gen_0/master.full.20040130234013"
2> go
WARNING: In order to LOAD the master database, the SQL Server must run in single-user mode. If
the master database dump uses multiple volumes, you must execute sp_volchanged on another SQL
Server at LOAD time in order to signal volume changes.
Backup Server session id is:  143.  Use this value when executing the 'sp_volchanged' system
stored procedure after fulfilling any volume change request from the Backup Server.
Backup Server: 6.28.1.1: Dumpfile name 'master0403014CDD ' section number 1 mounted on disk
file '/carin/sybbackups/science_pc/master/gen_0/master.full.20040130234013'
Backup Server: 4.58.1.1: Database master: 31748 kilobytes LOADed.
Backup Server: 4.58.1.1: Database master: 117770 kilobytes LOADed.
Backup Server: 4.58.1.1: Database master: 204808 kilobytes LOADed.
Backup Server: 4.58.1.1: Database master: 204818 kilobytes LOADed.
Backup Server: 3.42.1.1: LOAD is complete (database master).
02:00000:00006:2004/01/30 23:55:52.13 server  Logical Process Manager Error: Failed to use
database with id 1. Check preceding errors related to usability of this database.
(67 rows affected)
Msg 930, Level 14, State 1: Server 'science_pc', Line 1:
```

```
Database 'master' cannot be opened because either an earlier system termination left LOAD
DATABASE incomplete or the database is created with 'for load' option. Load the database or
contact a user with System Administrator (SA) role.
Logical Process Manager Error: Failed to use database with id 1. Check preceding errors
related to usability of this database.
Database 'master' is now online.
00:00000:00006:2004/01/30 23:55:52.59 server  Configuration file '/apps/licensed/sybase-
12.5/ASE-12_5/science_pc.cfg' has been written and the previous version has been renamed to
'/apps/licensed/sybase-12.5/ASE-12_5/science_pc.023'.
00:00000:00006:2004/01/30 23:55:52.63 server  The configuration option 'default sortorder id'
has been changed by 'sa' from '50' to '50'.
00:00000:00006:2004/01/30 23:55:52.63 kernel  ueshutdown: exiting
CT-LIBRARY error:       ct_results(): network packet layer: internal net library error: Net-
Library operation terminated
```

8.1.1.11 Restart the server in single-user mode

```
ssh -l sybase <host where Sybase is running>
cd $SYBASE/ASE-12_5/install
./startserver -fRUN_<server>_single
```

____

8.1.1.12 Check sysusages, sysdatabases and sysdevices in detail against printouts of those tables from before the device was lost. If you find discrepancies or if you otherwise suspect that devices may have been added or databases created/altered after the master backup you just loaded was made, refer to section 8.1.8 below, “Contingency: Databases or Devices Were Altered After Last Backup” for instructions on how to run disk reinit and refit.

____

8.1.1.13 If you suspect logins were added/altered after the backup you just loaded was made, refer to section 8.1.9 below, “Contingency: Logins Were Added After Last Backup”

____

8.1.1.14 Run dbcc’s on databases located on master device and check for error msgs

```
/home/sybase/sybdba/script/dbcc.csh <server> master.model.sybsystemdb
```

____

8.1.1.15 Run dbcc checkalloc on all the user databases

____

8.1.1.16 Backup the master database

```
/home/sybase/sybdba/script/save_master_info.csh <server>
```

____

8.1.1.17 Restart the server in multi-user mode

```
ssh -l sybase <host where Sybase is running>
/apps/licensed/sybase_local/etc/sybase_stop
/apps/licensed/sybase_local/etc/sybase_start
```

____

8.1.1.18 Enable any crontab jobs that were disabled earlier

____

8.1.1.19 Allow users to log in again to the sybase server by unlocking all logins as appropriate. Log in to the recovered sybase server as a system administrator and issue the following:

```
isql -U<user> -S<server>
select ‘exec sp_locklogin ‘ + name +’, ‘”unlock”’
      from master..syslogins where name not in (“sa”, ”carin”, “probe”)
```

execute the generated sql commands

```
select name from master..syslogins where status & 2 = 2
```

No logins should typically be returned

____

8.1.2 Recover the master database from backup when the Sybase won’t start

Use this procedure when you are unable to use the Sybase server. This assumes that the master device is still intact, only the master database is damaged. It further assumes that you have a valid backup of the database.

Estimated time to implement: 20-30 min + time for dbcc’s on user databases as deemed necessary

8.1.2.1 Disable any crontab jobs that might try to log in to this Sybase server once it is back up _______

8.1.2.2 If the server is somehow still running, dump the transaction logs in the user database to guard against possible data loss if the master recovery fails

```
/home/sybase/sybdba/script/tranlog_backup.csh <server>
```

____

8.1.2.3 If the server is somehow still running, generate up-to-date printouts of the various system tables. If not, locate the most recent printouts in the /sybackup1/carin/sybbackups/<server>/master/gen_0 directory. This will show you exactly what the master device looked like before the rebuild.

```
/home/sybase/sybdba/script/save_master_info.csh <server>
```

____

8.1.2.4 Shut down the Sybase server

```
ssh –l sybase <host where Sybase is running>
/apps/licensed/sybase_local/etc/sybase_stop
or, from the os-prompt, if that doesn't work
kill -9 <process-id>
```

____

8.1.2.5 Verify you have a valid configuration file, <server>.cfg, in the $SYBASE/ASE-12_5 directory. The number of devices-configuration parameter must be set sufficiently high to avoid losing user databases. The highest device number used can be found in the latest errorlog. Add 1 to account for device number 0 if you have to manually adjust this parameter in the configuration file. Recover the configuration file from OS-backup tape if necessary. See section section 9.2 below, "Restoring a file from ufsbackup on tape (OS backups run by SA)" _______

8.1.2.6 Make an OS-backup of the master device so you can start all over if the master recovery fails. Execute this step even if the Sybase server is unavailable and the master database/device corrupted. If the recovery fails you are probably better off restarting with this master device version. The master device is the –d parameter in the server's RUN_server-file. The backup will take up 1 GB of disk space. To restore, reverse the input and output files - if /of

```
ssh –l sybase <host where Sybase is running>
dd if=<device name>  of=<filename> bs=1024k
example: dd if=/dev/vx/rdsk/vol01_01 of=/home/sybase/mstr.sci.dd bs=1024k
```

____

8.1.2.7 Create the new master database by running dataserver with parameters as shown below. The dataserver command starts sybase server, tries to rebuild the master device and shuts down the dataserver again once done. There are two different scenarios. Start with the first, least intrusive, option and try the second if the master database build fails. If both scenarios fail and, unless there are contra-indicative msgs, you will have to rebuild the master device. If that is the case go to section 8.1.4 below, "Recovering the master device using backups of master database".

- Only the **data** in master db is corrupted, configuration area and allocation pages are intact. A large number of upgrade msgs are generated before the Sybase server shuts down again. If the configuration area is corrupted you'll get msgs to that effect and you'll need to try the next scenario

```
ssh sybase <host where Sybase is running>
$SYBASE/ASE-12_5/bin/dataserver –d <master device> –w master
example: $SYBASE/ASE-12_5/bin/dataserver –d /dev/vx/rdsk/vol01_01 –w master
```

- The **configuration area** is corrupted but the allocation pages are intact. A large number of upgrade msgs are generated before the Sybase server shuts down

```
ssh sybase <host where Sybase is running>
cd $SYBASE/ASE-12_5/bin
dataserver –d <master device> –w master –z <page size> –b <device size>
example: dataserver –d /dev/vx/rdsk/vol01_01 –w master –z 4K –b 1G
```

____

8.1.2.8 Start the Sybase server in single-user mode. Note that this is now a minimal server. All custom data, including the sa password and user databases, is invisible until the master backup has been loaded

```
ssh –l sybase <host for sybase server>
cd $SYBASE/ASE-12_5/install
```

cp RUN_<server> RUN_<server>_single

append –m at end of the file

./startserver –f RUN_<server>_single _______

8.1.2.9 Make the backup server known to Sybase

```
isql –Usa –P<blank> -S<server>
```

```
update master..sysservers
      set srvnetname = “<server>_back where srvname = “SYB_BACKUP”
(or, sp_addserver “SYB_BACKUP”, null, “<server>_back”)               _______
```

8.1.2.10 Start the backup server if it’s not running

```
cd $SYBASE/ASE-12_5/install
./startserver -f RUN_<server>_back                                   _______
```

8.1.2.11 Load the master backup from disk. If the desired backup file is no longer present in /sybackup1/carin/sybbackups/<server>/master/<gen> it may reside in the Sybase home directory, /home/sybase/sybsystem/<server>/master/<gen>. If that is the case, copy the backup file to the /sybackup1/carin/sybbackups directory, as the load script requires. If the backup file cannot be found in either directory, load it from an os-system backup tape, see section 9.2 below, “Restoring a file from ufsbackup on tape (OS backups run by SA)”. The Sybase server shuts down after the load is complete

```
isql -Usa -P<blank> -S<server>
/* find the exact name of the backup file */
!! ls -l /home/sybase/sybsystem/<server>/master/gen_0/master.full*
load database master from
  “/home/sybase/sybsystem/<server>/master/gen_0/master.full.<date>”______
```

8.1.2.12 Check for error messages in the output from the isql-session and in the errorlog $SYBASE/ASE-12_5/install/<server>.log. You can safely ignore error msg 930, failing to open the master database, as long as there is a msg stating that the master database is online further down. The following sample output signifies a successful master db load:

```
1> load database master from
"/carin/sybbackups/science_pc/master/gen_0/master.full.20040130234013"
2> go
WARNING: In order to LOAD the master database, the SQL Server must run in single-user mode. If
the master database dump uses multiple volumes, you must execute sp_volchanged on another SQL
Server at LOAD time in order to signal volume changes.
Backup Server session id is:  143.  Use this value when executing the 'sp_volchanged' system
stored procedure after fulfilling any volume change request from the Backup Server.
Backup Server: 6.28.1.1: Dumpfile name 'master0403014CDD ' section number 1 mounted on disk
file '/carin/sybbackups/science_pc/master/gen_0/master.full.20040130234013'
Backup Server: 4.58.1.1: Database master: 31748 kilobytes LOADed.
Backup Server: 4.58.1.1: Database master: 117770 kilobytes LOADed.
Backup Server: 4.58.1.1: Database master: 204808 kilobytes LOADed.
Backup Server: 4.58.1.1: Database master: 204818 kilobytes LOADed.
Backup Server: 3.42.1.1: LOAD is complete (database master).
02:00000:00006:2004/01/30 23:55:52.13 server  Logical Process Manager Error: Failed to use
database with id 1. Check preceding errors related to usability of this database.
(67 rows affected)
Msg 930, Level 14, State 1: Server 'science_pc', Line 1:
Database 'master' cannot be opened because either an earlier system termination left LOAD
DATABASE incomplete or the database is created with 'for load' option. Load the database or
contact a user with System Administrator (SA) role.
Logical Process Manager Error: Failed to use database with id 1. Check preceding errors
related to usability of this database.
Database 'master' is now online.
00:00000:00006:2004/01/30 23:55:52.59 server  Configuration file '/apps/licensed/sybase-
12.5/ASE-12_5/science_pc.cfg' has been written and the previous version has been renamed to
'/apps/licensed/sybase-12.5/ASE-12_5/science_pc.023'.
00:00000:00006:2004/01/30 23:55:52.63 server  The configuration option 'default sortorder id'
has been changed by 'sa' from '50' to '50'.
00:00000:00006:2004/01/30 23:55:52.63 kernel  ueshutdown: exiting
CT-LIBRARY error: ct_results(): network packet layer: internal net library error: Net-Library
operation terminated
```

8.1.2.13 Restart the server in single-user mode

```
ssh -l sybase <host where Sybase is running>
cd $SYBASE/ASE-12_5/install
./startserver -fRUN_<server>_single                                  _______
```

8.1.2.14 Check sysusages, sysdatabases and sysdevices in detail against printouts of those tables from before the device was lost. If you find discrepancies or if you otherwise suspect that devices may have been added or databases created/altered after the master backup you just loaded was made, refer to section 8.1.8 below, “Contingency: Databases or Devices Were Altered After Last Backup” for instructions on how to run disk reinit and refit. ____

8.1.2.15 If you suspect logins were added/altered after the backup you just loaded was made, refer to section 8.1.9 below, “Contingency: Logins Were Added After Last Backup” __________

8.1.2.16 Run dbcc’s on databases located on master device and check for error msgs

```
/home/sybase/sybdba/script/dbcc.csh <server> master.model.sybsystemdb
```

____

8.1.2.17 Run dbcc checkalloc on all the user databases ____

8.1.2.18 Backup the master database

```
/home/sybase/sybdba/script/save_master_info.csh <server>
```

____

8.1.2.19 Restart the server in multi-user mode

```
ssh -l sybase <host where Sybase is running>
/apps/licensed/sybase_local/etc/sybase_stop
/apps/licensed/sybase_local/etc/sybase_start
```

____

8.1.2.20 Enable any crontab jobs that were disabled earlier __________

8.1.3 Recover the master database when there is no master backup

When there is no valid master database backup, you must rebuild the entire master device using the instructions in section 8.1.5 below, “Rebuild the master device when there is no master backup”.

8.1.4 Recovering the master device using backups of master database

If not only the master database but also the underlying device is corrupted, the procedure outlined in this section should be followed. This is a critical undertaking with potential for serious problems if it is not done correctly. It is assumed that a valid backup of the master database itself exists.

Estimated time to implement: 40-60 min + time for dbcc’s on user databases

8.1.4.1 Disable any crontab jobs that might try to log in to this Sybase server _______

8.1.4.2 If the server is somehow still running, dump the transaction logs in the user database to guard against possible data loss if the master recovery fails

```
/home/sybase/sybdba/script/tranlog_backup.csh <server>
```

____

8.1.4.3 Generate up-to-date printouts of the various system tables if the server is still running. If not, locate the most recent printouts in the /sybackup1/carin/sybbackups/<server>/master/gen_0 directory. This will show you exactly what the master device looked like before the rebuild.

```
/home/sybase/sybdba/script/save_master_info.csh <server>
```

____

8.1.4.4 Verify you have a valid configuration file, <server>.cfg, in the $SYBASE/ASE-12_5 directory. The number of devices-configuration parameter must be set sufficiently high to avoid losing user databases. The highest device number used can be found in the latest errorlog. Add 1 to account for device number 0 if you have to manually adjust this parameter in the configuration file. Recover the configuration file from the nightly OS-backups if necessary (section 9.2 below, “Restoring a file from ufsbackup on tape (OS backups run by SA)”) _______

8.1.4.5 Shut down the Sybase server

```
ssh -l sybase <host where sybase is running>
/apps/licensed/sybase_local/etc/sybase_stop
or, from the os-prompt, if that doesn’t work
kill -9 <process-id>
```

____

8.1.4.6 Make an OS-backup of the master device so you can start all over if the master recovery fails. The master device is the –d parameter in the server's RUN_server-file. The backup will take up as much space as the device being backed up, 1GB. To restore, reverse the input and output files - if /of

```
ssh –l sybase <host where Sybase is running>
dd if=<device name>  of=<filename> bs=1024k
example: dd if=/dev/vx/rdsk/vol01_01 of=/home/sybase/mstr.sci.dd bs=1024k
```
____

8.1.4.7 Any databases other than master, model, tempdb or sybsystemdb that are fully or partially located on the master device will be destroyed when the master device is rebuilt. They will have to be recreated and reloaded later. Do **not** put them back on the master device _______

8.1.4.8 Decide where to build the new master device in case you decide to relocate to a new device. The new device must be at least as large as the existing one _______

8.1.4.9 Create the new master device. If rebuilding on top of an existing device, add –f parameter to force the build. After generating multiple upgrade messages the Sybase server shuts down

```
cd $SYBASE/ASE-12_5/bin
dataserver –d <master device> –z <page size> –b <device size>
example: dataserver –d /dev/vx/rdsk/vol01_01 –z4K –b1024M –f
```
____

8.1.4.10 If you relocated the master device, adjust the RUN_<server> file accordingly (-d param) ____

8.1.4.11 Start the Sybase server in single-user mode. Note that this is now a minimal server. All custom data, including the sa password and user databases, is invisible until the master backup is loaded.

```
ssh –l sybase <host for sybase server>    # sci-base/sci-crunch
cd $SYBASE/ASE-12_5/install
cp RUN_<server> RUN_<server>_single
append –m at end of the file
./startserver  –f RUN_<server>_single
```
____

8.1.4.12 Expand the master database so the dump will fit. To avoid problems down the road, master, model, tempdb and sybsystemdb allocations on the master device must be recreated **exactly** as before the rebuild. No other databases should exist on this device. If this is all true simply follow the instructions in this step. If not, refer to step 4 in Chapter 28 in the Sybase System Administration Guide.

- Take out the printouts located in step 8.1.4.3 above, they show the desired final outcome
- Find the "high"-column in the sysdevices printout for device "master"
- Now locate the sysusages printout that is ordered by vstart. All entries where vstart < "high" from previous step show master device-entries. It should look something like this:

```
dbid   segmap      lstart      size        vstart
------ ----------- ----------- ----------- -----------
     1           7           0        3072           4
     3           7           0        1024        6148
     2           7           0        1024        8196
 31513           7           0        1024       10244
     1           7        3072       48128       12292
```

- Compare that to the output on the newly rebuilt server:

```
isql –Usa –P<blank> –S<server> –w222
select db_name(dbid), dbid, segmap, lstart, size, vstart
from   sysusages order by vstart

             dbid   segmap      lstart      size        vstart
------------ ------ ----------- ----------- ----------- -----------
master            1           7           0        3072           4
tempdb            3           7           0        1024        6148
model             2           7           0        1024        8196
sybsystemdb   31513           7           0        1024       10244
```

- In this particular example the master database needs to be expanded per the last row in the first output above. “Size” is number of pages; to get MB, divide by 256. Then expand the database accordingly:

```
alter database master on master = 188                            _______
```

8.1.4.13 Make the backup server known to Sybase

```
isql –Usa –P<blank> -S<server>
update master..sysservers
      set srvnetname = “<server>_back where srvname = “SYB_BACKUP”
(or, sp_addserver “SYB_BACKUP”, null, “<server>_back”)           _______
```

8.1.4.14 Start the backup server if it’s not running

```
$SYBASE/ASE-12_5/install/startserver –f RUN_<server>_back        _______
```

8.1.4.15 Load the master backup from disk. If the desired backup file is no longer present in /sybackup1/cairn/sybbackups/<server>/master/<gen> directory, it may still reside in the /home/sybase/sybsystem/<server>/master/<gen> directory. If that is the case, copy the backup file to the /sybackup1/carin/sybbackups directory, this is what the load script expects. If the backup file cannot be found in either directory, load it from an os-system backup tape, see section 9.2 below, “Restoring a file from ufsbackup on tape (OS backups run by SA)”. The Sybase server shuts down after the load is complete

```
isql –Usa –P<blank> -S<server>
/* find the exact name of the backup file */
!! ls -l /home/sybase/sybdba/sybbackups/<server>/master/gen_0/master.full*
load database master from
  “/home/sybase/sybdba/sybbackups/<server>/master/gen_0/master.full.<date>”___
```

8.1.4.16 Check for error messages in the output from the isql-session and in the errorlog $SYBASE/ASE-12_5/install/<server>.log. You can safely ignore errors such as 953 and 930 below. The following sample output signifies a successful master db load:

```
1> load database master from
"/home/sybase/sybdba/sybbackups/science_pc/master/gen_0/master.full.20040130234013"
2> go
WARNING: In order to LOAD the master database, the SQL Server must run in single-user mode.
If the master database dump uses multiple volumes, you must execute sp_volchanged on another
SQL Server at LOAD time in order to
signal volume changes.
Backup Server session id is:  124.  Use this value when executing the 'sp_volchanged' system
stored procedure after fulfilling any volume change request from the Backup Server.
Backup Server: 6.28.1.1: Dumpfile name 'master0403014CDD ' section number 1 mounted on disk
file '/home/sybase/sybdba/sybbackups/science_pc/master/gen_0/master.full.20040130234013'
Backup Server: 4.58.1.1: Database master: 31748 kilobytes LOADed.
Backup Server: 4.58.1.1: Database master: 117770 kilobytes LOADed.
Backup Server: 4.58.1.1: Database master: 204808 kilobytes LOADed.
Backup Server: 4.58.1.1: Database master: 204818 kilobytes LOADed.
Backup Server: 3.42.1.1: LOAD is complete (database master).
Msg 953, Level 11, State 1:
Line 1:
Page '1024', passed to curunreservedpgs built-in function, is an invalid page number in
database ID '2'. The highest page number in this database is '1023'.
Msg 953, Level 11, State 1:
Line 1:
Page '1311744', passed to curunreservedpgs built-in function, is an invalid page number in
database ID '2'. The highest page number in this database is '1023'.
02:00000:00006:2004/02/05 23:18:23.88 server  Logical Process Manager Error: Failed to use
database with id 1. Check preceding errors related to usability of this database.
Msg 953, Level 11, State 1:
Line 1:
Page '2622464', passed to curunreservedpgs built-in function, is an invalid page number in
database ID '2'. The highest page number in this database is '1023'.
(68 rows affected)
Msg 930, Level 14, State 1:
Line 1:
Database 'master' cannot be opened because either an earlier system termination left LOAD
DATABASE incomplete or the database is created with 'for load' option. Load the database or
contact a user with System Administrator
(SA) role.
Logical Process Manager Error: Failed to use database with id 1. Check preceding errors
related to usability of this database.
00:00000:00006:2004/02/05 23:18:24.34 server  Configuration file '/apps/licensed/sybase-
12.5/ASE-12_5/science_pc.cfg' has been written and the previous version has been renamed to
'/apps/licensed/sybase-12.5/ASE-12_5/science_pc.029'.
```

```
00:00000:00006:2004/02/05 23:18:24.39 server  The configuration option 'default sortorder id'
has been changed by 'sa' from '50' to '50'.
00:00000:00006:2004/02/05 23:18:24.40 kernel  ueshutdown: exiting
CT-LIBRARY error:
        ct_results(): network packet layer: internal net library error: Net-Library operation
terminated due to disconnect
```

8.1.4.17 Restart the server in single-user mode. All the user databases should now be recovered

```
ssh -l sybase <host for sybase server>    # sci-base/sci-crunch
cd $SYBASE/ASE-12_5/install
./startserver -fRUN_<server>_single
```

____

8.1.4.18 The privileges on model and sybsystemdb are lost during the rebuild, recover both from backups

```
sp_configure “enable xact coordination”, 0      -- to load sybsystemdb
<recycle sybase server, starting server in multi-user>
!!ls -l /home/sybase/sybsystem/<server>/model/gen_0/model.full.*
load database model from “<filename>”
online database model
!!ls -l home/sybase/sybsystem/<server>/sybsystemedb/gen_0/*full.*
load database sybsystemdb from “<filename>”
online database sybsystemdb
sp_configure “enable xact coordination”, 1      -- reset after working on
sybsystemdb
<recycle sybase server, this reinitializes tempdb as well>
```

____

8.1.4.19 If there were additional databases on the master device, drop and recreate those databases and recover from backup. Do **not** put any section back on the master device

____

8.1.4.20 Check sysusages, sysdatabases and sysdevices in detail against printouts of those tables from before the device was lost. If you find discrepancies or if you otherwise suspect that devices may have been added or databases created/altered after the master backup you just loaded was made, refer to section 8.1.8 below, “Contingency: Databases or Devices Were Altered After Last Backup” for instructions on how to run disk reinit and refit.

____

8.1.4.21 If you suspect logins were added/altered after the backup you just loaded was made, refer to section 8.1.9 below, “Contingency: Logins Were Added After Last Backup”

____

8.1.4.22 Run full dbcc’s on databases located on master device and check for error msgs

```
/home/sybase/sybdba/script/dbcc.csh <server> master.model.sybsystemdb
```

____

8.1.4.23 Run dbcc checkalloc on each system and user database

____

8.1.4.24 Backup the master database

```
/home/sybase/sybdba/script/save_master_info.csh <server>
```

____

8.1.4.25 Restart the server in multi-user mode

```
ssh -l sybase <host where Sybase is running>
/apps/licensed/sybase_local/etc/sybase_stop
/apps/licensed/sybase_local/etc/sybase_start
```

____

8.1.4.26 Enable any cronjobs that were previously disabled

____

8.1.5 Rebuild the master device when there is no master backup

The following procedure rebuilds the master device and reconstructs the master database using copies of the critical system tables. It is important to really understand all the steps and their implications. Changes made incorrectly to the master tables may render the user databases unusable.

Estimated time to implement: 40-60 min + time for dbcc’s on user databases

8.1.5.1 Disable any crontab jobs that might try to log in to this Sybase server _______

8.1.5.2 If the server is still running, dump the transaction logs in the user database to guard against possible data loss if the master recovery fails

```
/home/sybase/sybdba/script/tranlog_backup.csh <server>
```

____

8.1.5.3 Generate up-to-date printouts and bcp-files of the various system tables if the server is still somehow running. If not, locate the most recent printouts and bcp-files of system tables in the /sybackup1/carin/sybbackups/<server>/master/gen_0 directory. This will show you exactly what the master device looked like before the rebuild. Even if this data does not reflect how the master database should be rebuilt, the information will prove useful in reconstructing db allocations etc

```
/home/sybase/sybdba/script/save_master_info.csh <server>
```

____

8.1.5.4 Locate the set of printouts and bcp-files to use for building the master device/database. Files are located in /sybackup1/carin/sybbackups/<server>/master/gen_0 directory. _______

8.1.5.5 Verify you have a valid configuration file, <server>.cfg, in the $SYBASE/ASE-12_5 directory. The number of devices-configuration parameter must be set sufficiently high to avoid losing user databases. The highest device number used can be found in the latest errorlog. Add 1 to account for device number 0 if you have to manually adjust this parameter in the configuration file. Recover the configuration file from the nightly OS-backups if necessary (section 9.2 below, “Restoring a file from ufsbackup on tape (OS backups run by SA)”) _______

8.1.5.6 Shut down the Sybase server if it is still running

```
ssh -l sybase <host where sybase is running>
/apps/licensed/sybase_local/etc/sybase_stop
or, from the os-prompt, if that doesn’t work
kill -9 <process-id>
```

____

8.1.5.7 Make an OS-backup of the master device so you can start all over if the master recovery fails. The master device is the –d parameter in the server’s RUN_server-file. The backup will take up as much space as the device being backed up, 1GB. To restore, reverse the input and output files - if /of

```
ssh -l sybase <host where Sybase is running>
dd if=<device name>  of=<filename> bs=1024k
example: dd if=/dev/vx/rdsk/vol01_01 of=/home/sybase/mstr.sci.dd bs=1024k
```

____

8.1.5.8 Any databases other than master, model, tempdb or sybsystemdb that are fully or partially located on the master device will be destroyed when the master device is rebuilt. They will have to be recreated and reloaded later. Do **not** put them back on the master device _______

8.1.5.9 Decide where to build the new master device in case you decide to relocate to a new device. The new device must be at least as large as the existing one _______

8.1.5.10 Create the new master device. If rebuilding on top of an existing device, add –f parameter to force the build. After generating multiple upgrade messages the Sybase server shuts down

```
cd $SYBASE/ASE-12_5/bin
dataserver -d <master device> -z <page size> -b <device size>
example: dataserver -d /dev/vx/rdsk/vol01_01 -z4K -b1024M -f
```

____

8.1.5.11 If you relocated the master device, adjust the RUN_<server> file accordingly (-d param) ____

8.1.5.12 Start the Sybase server in single-user mode. Note that this is now a minimal server. All custom data, including the sa password and user databases, is invisible until the master backup is loaded.

```
ssh -l sybase <host where Sybase is running>
cd $SYBASE/ASE-12_5/install
```

cp RUN_<server> RUN_<server>_single

append –m at end of the file

./startserver –f RUN_<server>_single _______

8.1.5.13 Temporarily add proc sp_serverinfo to the master database. Get the source code from $SYBASE/ASE-12_5/scripts/installmaster. Don’t worry about adding sp_getmessage

____

8.1.5.14 Adjust sysusages bcp file to remove entries for master, model, tempdb and sybsystemdb databases (so as not to use uninitialized space on the master device or creating gaps in lstarts in sysusages)

```
cd /home/sybase/sybdba/sybbackups/<server>/master/<gen>
cp sysusages.<server>.c sysusages.<server>.c.orig
<remove rows where first column is 1,2,3 or 31513>
```

8.1.5.15 If databases other than master, model, tempdb and sybsystemdb are located on master device, remove those entries from the sysusages bcp file as well

isql –Usa –P<blank> -S<server>

select high from master..sysdevices where name = ‘master’

<remove rows from sysusages.<server>.c where vstart < high from previous query>

8.1.5.16 Delete configuration table

```
isql –Usa –P<blank> –S<server> –w222
delete master..sysconfigures
```

8.1.5.17 Bcp in the saved off system tables. –b continues processing when errors are encountered. Use the character mode –c for all bcps except syscharsets, where –n, native is used.

```
cd /home/sybase/sybdba/sybbackups/<server>/master/<gen>
bcp master..sysusages in sysusages.<server>.c –Usa –P –S<srv> -b 1 –c
bcp master..sysdevices in sysdevices.<server>.c –Usa –P –S<srv> -b 1 –c
bcp master..sysdatabases in sysdatabases.<server>.c –Usa –P –S<srv> -b 1 –c
bcp master..sysservers in sysservers.<server>.c –Usa –P –S<srv> -b 1 –c
bcp master..syslogins in syslogins.<server>.c –Usa –P –S<srv> -b 1 –c
bcp master..sysloginroles in sysloginroles.<server>.c –Usa –P –S<srv> -b 1 –c
bcp master..sysusers in sysusers.<server>.c –Usa –P –S<srv> -b 1 –c
bcp master..sysconfigures in sysconfigures.<server>.c –Usa –P –S<srv> -b 1 –c
bcp master..syscharsets in syscharsets.<server>.c –Usa –P –S<srv> -b 1 –n
bcp master..sysremotelogins in sysremotelogins.<server>.c –Usa –P –S<srv> -b 1 –c
bcp master..sysresourcelimits in sysresourcelimits.<server>.c –Usa –P –S<srv> -b 1 –c
bcp master..systimeranges in systimeranges.<server>.c –Usa –P –S<srv> -b1 –c
Check for errors
```

8.1.5.18 Recycle the server, staying in single-user mode. Use shutdown with nowait to avoid unnecessary errors. All the user databases should be recovered when the Sybase server starts

```
isql –Usa –P<password> –S<server>
shutdown with nowait
ssh –l sybase <host for sybase server>    # sci-base/sci-crunch
cd $SYBASE/ASE-12_5/install
./startserver –fRUN_<server>_single
```

8.1.5.19 Ignore error 7727, “Invalid role string ‘mon_role’ entered”, it will be fixed later on

8.1.5.20 Verify in the errorlog that all looks ok and all databases have been recovered

8.1.5.21 If there were additional databases on the master device, drop those databases. Do **not** put any section back on the master device

8.1.5.22 Verify that the user and system databases seem OK. Ignore master, model, tempdb and sybsystemdb for now

```
isql –Usa –P<password> –S<server>
declare @pgspermb int
select  @pgspermb = 1048576/@maxpagesize
select  “db”=db_name(dbid), dbidb, “mb”=sum(size_/@pgspermb
from master..sysusages group by dbid
```

8.1.5.23 If there are orphaned rows, remove them

```
isql –Usa –P<password> –S<server>
-- locate & delete orphans in sysusages
select * from master..sysusages where db_name(dbid) = null  -- locate
orphans in sysusages
go
begin tran
  delete master..sysusages where db_name(dbid) = null  -- delete orphans
  -- was the correct number of rows deleted?
go
commit tran/rollback tran
go
-- locate & delete orphans in sysdatabases
select dbid, name from sysdatabases where not exists
  (select * from sysusages where dbid = sysdatabases.dbid)
go
begin tran
  select dbid, name from sysdatabases where not exists
    (select * from sysusages where dbid = sysdatabases.dbid)
  -- was the correct number of rows deleted?
go
commit tran/rollback tran
go
```

8.1.5.24 Expand master and tempdb databases to their original sizes. (If model or sybsystemdb were expanded beyond 4mb, expand them too)

```
alter database master on master = 188
alter database tempdb on dev_TMP_a=’5G’, dev_TMP_b=’5G’
     log on dev_TMP_log_a=‘5G’
```

8.1.5.25 If any database is online for standby_access bring them online. It is necessary for all databases to be writable for the installmaster script to install mon_role in master properly

```
online database <database> etc.
```

8.1.5.26 Run the installation scripts for master, model etc

```
cd $SYBASE/ASE-12_5/scripts
isql -Usa -P<pwrd> -S<server> <installmaster >installmaster.<server>.<date>
isql -Usa -P<pwrd> -S<server> <installmodel >installmodel   .<server>.<date>
isql -Usa -P<pwrd> -S<server> <instmsgs.ebf >instmsgs.<server>.<date>
isql -Usa -P<pwrd> -S<server> <installjconnect >installjconnect.<server>.<date>
isql -Usa -P<pwrd> -S<server> <installdbccdb >installdbccdb.<server>.<date>
check for error msgs
```

8.1.5.27 Drop the procedure sp_serverinfo from the master database

```
use master
drop procedure sp_serverinfo
```

8.1.5.28 Recreate and recover from backup any databases that were deleted because they were fully or partially located on the master device. Do **not** allocate db on the master device again

8.1.5.29 Check sysusages, sysdatabases and sysdevices in detail against printouts of those tables from before the device was lost. If you find discrepancies or if you otherwise suspect that devices may have been added or databases created or altered on this server after the bcp files were created, refer to section 8.1.8 below for instructions on how to run disk reinit and refit. **Do not simply reissue the disk init or create/alter database commands!**

8.1.5.30 If you suspect logins have been added, deleted, locked or in any way changed since the bcp-files were generated, refer to section 8.1.9 below for instructions on how to fix this problem

8.1.5.31 Run full dbcc’s on databases located on master device and check for error msgs

```
/home/sybase/sybdba/script/dbcc.csh <server> master.model.sybsystemdb
```

8.1.5.32 Run dbcc checkalloc on each system and user database

8.1.5.33 Backup the master database

```
/home/sybase/sybdba/script/save_master_info.csh <server>
```

8.1.5.34 Restart the server in multi-user mode

```
ssh –l sybase <host where Sybase is running>
/apps/licensed/sybase_local/etc/sybase_stop
/apps/licensed/sybase_local/etc/sybase_start
```

8.1.5.35 Enable any cronjobs that were previously disabled

8.1.6 Contingency: Recover master backup files from backup disk before loading

The last three generations of master backups can be found in the /home/sybase/sybsystem directory.

8.1.6.1 Copy the files you need to the appropriate /sybackup1/carin/sybbackups directory. Scripts expect files to be located in the /sybackup1/carin/sybbackups directory.

```
cd /sybacku1/carin/sybbackups/<server>/master/<gen>
cp /home/sybase/sybsystem/<server>/master/<gen>/<file> .
```

8.1.6.2 Continue with recovery where you left off in the writeups above.

8.1.7 Contingency: Recover master backup files from OS backup tape before loading

8.1.7.1 The master backup files get backed up onto the OS backup tapes nightly. Use procedure in section 9.2 below, “Restoring a file from ufsbackup on tape (OS backups run by SA)” to load a particular file from tape to disk

8.1.7.2 Continue with recovery where you left off in the writeups above.

8.1.8 Contingency: Databases or Devices Were Altered After Last Backup

If devices were added or databases expanded or added after the last valid backup was made, these changes are not reflected in the system tables in the master database after the backup is loaded. The procedures below updates the system tables with the changes already made and written to disk. Failure to go through these steps if changes indeed were made may result in data loss and corrupted user databases.

Estimated time to implement: 30-60 minutes + time for full dbcc’s

8.1.8.1 Start the Sybase server in single-user mode and with trace flag 3608

```
ssh –l sybase <server>
cd $SYBASE/ASE-12_5/install
cp RUN_<server> RUN_<server>_T3608
edit RUN_<server>_T3608, add –m –T3608 at the end of the dataserver command
shutdown the Sybase server if it’s running
./startserver –f RUN_<server>_T3608
```

8.1.8.2 Check if database devices were added after last valid backup by reviewing previous dataserver logs. Were any devices started that are currently not active? Were any devices initialized while not starting the Sybase server (see example below)? Was email sent out indicating disk or database changes between production and warm standby server?

```
select name, phyname from master..sysdevices  -- lists current db devices

Example of a new device being added as show in dataserver log:
02:00000:00040:2004/02/09 01:50:08.80 server  Checking space allocation for database 20
02:00000:00040:2004/02/09 01:50:08.80 server  DBCC TRACEON 3605, SPID 40
02:00000:00040:2004/02/09 01:50:08.86 server  DBCC TRACEOFF 3605, SPID 40
02:00000:00041:2004/02/09 17:27:33.76 kernel  Initializing virtual device 51,
'/dev/vx/rdsk/vol01_13' with dsync 'on'.
02:00000:00041:2004/02/09 17:27:33.77 kernel  Virtual device 51 started using
asynchronous i/o.
```

8.1.8.3 Reinitialize the database devices in question. (Do not run a regular disk init, if the device is actively being used, data on it will be lost). If you do not have any scripts showing how the disk inits in question was run, you may cull that information from the dataserver log

```
isql –Usa –P<password> -S<server>
disk reinit name = "<logical name>", physname = "<raw device name>",
      size = "<size>"
```

8.1.8.4 If databases were altered or added after the last valid backup a disk refit must be run. The disk refit adjusts sysdatabases and sysusages according to what it finds on the disks themselves. The Sybase server shuts down when the command finishes

```
isql –Usa –P<password> -S<server>
disk refit
```

8.1.8.5 Restart the Sybase server without traceflags. Shut down server with nowait if it is still running

```
ssh –l sybase <host where server resides>
isql –Usa –P<password> -S<server>
shutdown with nowait
cd $SYBASE/ASE-12_5/install
./startserver –f RUN_<server>
```

8.1.8.6 Run queries to find orphaned rows

```
isql –Usa –P<password> -S<server>
-- locate & delete orphans in sysusages
select * from master..sysusages where db_name(dbid) = null  -- locate
orphans in sysusages
go
begin tran
  delete master..sysusages where db_name(dbid) = null  -- delete orphans
  -- was the correct number of rows deleted?
go
commit tran/rollback tran
go
-- locate & delete orphans in sysdatabases
select dbid, name from sysdatabases where not exists
  (select * from sysusages where dbid = sysdatabases.dbid)
go
begin tran
  select dbid, name from sysdatabases where not exists
    (select * from sysusages where dbid = sysdatabases.dbid)
  -- was the correct number of rows deleted?
go
commit tran/rollback tran
go
```

8.1.8.7 Compare disk inits and create_db commands for the two servers. It should give you a clue as to whether you've succeeded in comptetely recovering the failed system.

```
/home/sybase/sybdba/script/cmp_create_db.csh science_prod science_pc
more /home/sybase/sybdba/wrk/create_db_for_load.science_prod.science_pc.diff
/home/sybase/sybdba/script/cmp_disk_init.csh science_prod science_pc
more /home/sybase/sybdba/wrk/disk_init.science_prod.science_pc.diff
```

8.1.8.8 Run checkalloc dbccs on all system and user databases. (If time allows, run complete dbcc's)

```
dbcc checkalloc (<database>)  etc
```

8.1.9 Contingency: Logins Were Added After Last Backup

Run this query if you suspect logins were added after last valid backup.

Estimated time to implement: 10 minutes

8.1.9.1 Check if there are any orphaned logins in any user database

```
select “select suid,name from “+name+”..sysusers where suid  > 0 and not
exists (select * from master..syslogins L where suid=L.suid)”
<execute all the generated queries>
```

____

8.1.9.2 If any user ids were displayed in the previous query, these are the orphaned logins. Run sp_addlogin to add the orphaned rows. Note that it is imperative that the suid is retained. You may have to temporarily create dummy-logins to make sure you get the same suid.

8.1.9.3 It is not possible to identify logins that were dropped, changed or locked/unlocked after the last backup. Run a comparison between the production and standby servers and compare the differences. You should be able to figure out if further manual changes need be made to the recovered system

```
/home/sybase/sybdba/script/cmp_logins.csh science_prod science_pc
more /home/sybase/sybdba/wrk/logins.science_prod.science_pc.diff
```

____

8.2 Recovering the model Database

The model database has not been customized for the GPB installation, which simplifies the recovery procedure. The database cannot be explicitly dropped. It must be rebuilt using the dataserver command as described in section 8.2.3 below. The model database must reside on the master device; if it is damaged, rebuild it per the instructions in section 8.1 above.

8.2.1 Using Regular Disk Backup

Estimated time to implement: 5-10 min

8.2.1.1 Load the backup. Since the database has not been customized for the GPB installation, you can use the science_prod backup for recovery on science_pc and vice versa. The generation of the backup file on disk is also irrelevant

```
isql –U<user> –S<server>
load database model from
 ”/home/sybase/sybsystem/<server>/model/<generation>/model.full.<date>”
go
online database model
go
```

____

8.2.1.2 Verify model looks ok. You can use it and there are 26 rows in sysobjects/sysprotects _______

8.2.1.3 Restart the server to make sure that tempdb get initialized correctly

```
ssh –l sybase <host where Sybase is running>
/apps/licensed/sybase_local/etc/sybase_stop
/apps/licensed/sybase_local/etc/sybase_start
```

____

8.2.1.4 Backup model database

```
/home/sybase/sybdba/script/full_backup.csh <server> model
```

____

8.2.2 Contingency 1: Backup a working model database and use for recovery

Estimated time to implement: 5-10 min.

8.2.2.1 Dump the model database from a working Sybase server, **not** the one you are working on

```
/home/sybase/sybdba/script/full_backup.csh <working_server> model
```

____

8.2.2.2 Continue with recovery per section 8.2.1 above. Adjust the path name to the backup file _______

8.2.3 Contingency 2: Rebuild the model database using dataserver utility

Estimated time to implement: 5-10 min.

The instructions in Chapter 28 in Sybase System Administration Guide to use option –x are incorrect.

8.2.4 ssh to the host that the dataserver is running from. This is CRUCIAL.

```
ssh –l sybase sci-base    # use for science_prod database server
ssh –l sybase sci-crunch  # use for science_pc database server
```

____

8.2.4.1 Shut down the dataserver gracefully _______

8.2.4.2 Verify the name of the master device in the appropriate RUN-server file (-d parameter)

Science_prod: –d /dev/vx/rdsk/vol01_01

Science_pc: –d /dev/vx/rdsk/vol01_01 _______

8.2.4.3 Build a new model database from scratch. Adjust the –d parameter per step 8.2.4.2 above. The server shuts down automatically when model has been built. Sample dataserver command:

```
$SYBASE/ASE-12.5/bin/dataserver –d /dev/vx/rdsk/vol01_01 –w model
```

____

8.2.4.4 Restart the dataserver _______

8.2.4.5 Install grants on system tables

cd $SYBASE/ASE-12.5/scripts

```
isql –U<user> –P<pwd> –S<svr> <installmodel >installmodel.<svr>.<date>
```

Check the installmodel.<svr>.<date> file for error msgs _______

8.2.4.6 Verify model looks ok. You can use it and there are 26 rows in sysobjects/sysprotects _______

8.2.4.7 Restart the server again to make sure that tempdb get initialized correctly __________

8.2.4.8 Backup master and model databases

```
/home/sybase/sybdba/script/save_master_info.csh <server>
/home/sybase/sybdba/script/full_backup.csh <server> model
```

____

Contingency 3: Recovering model database backups from tape
Estimated time to implement: 1 hr

8.2.4.9 The model backup file gets backed up onto the OS backup tapes nightly. Use procedure in section 9.2 below, “Restoring a file from ufsbackup on tape (OS backups run by SA)” to load backup file from tape to disk _______

8.2.4.10 Continue with recovery per section 8.2.1 above. Adjust the path name to the backup file ____

8.3 Recovering the sybsystemprocs Database

The sybsystemprocs database has been customized for the GPB installation. It is not sufficient to simply reinstall the generic version.

8.3.1 Using Regular Disk Backup

Estimated time to implement: 5-10 min

8.3.1.1 Load the backup. Although the database has been customized for the GPB installation, you can use the science_prod backup for recovery on science_pc and vice versa. The backup generation may matter if stored procs have been modified.

```
isql –U<user> –S<server>
load database sybsystemprocs from
“/home/sybase/sybsystem/<server>/sybsystemprocs/gen_0/sybsystemprocs.full.<date>”
go
online database sybsystemprocs
go
```

____

8.3.1.2 Verify sybsystemprocs looks ok. _______

8.3.1.3 Backup sybsystemprocs database

```
/home/sybase/sybdba/script/full_backup.csh <server> sybsystemprocs
```

____

8.3.2 Contingency 1: Backup a working sybsystemprocs database and use for recovery

Estimated time to implement: 5-20 min.

8.3.2.1 Dump the sybsystemprocs database from a working Sybase server, **not** the one you are working on

```
/home/sybase/sybdba/script/full_backup.csh <work’n_server> sybsystemprocs
```

____

8.3.2.2 Continue with recovery per section 8.3.1 above. Adjust the path name to the backup file _______

8.3.3 Contingency 2: Rebuild the sybsystemprocs database from scratch

Estimated time to implement: 20-25 min.

8.3.4 Backup the master database since objects in it will be updated when installmaster/msgs are run

```
/home/sybase/sybdba/script/save_master_info.csh <server>
```

____

8.3.5 Drop and recreate the sybsystemprocs database**.** Do **not** create the database with the for load option. This is recommended to ensure that the sysprocedures table does not grow unnecessarily when installmaster is run multiple times. Use the allocations as noted in the script /sybackup1/carin/sybbackups/<server>/master/gen_0/create_db_for_load.<server>.csh. If the file is missing, use the one in the /home/sybase/sybsystem directory or recover from the OS-backup per section 9.2 below, “Restoring a file from ufsbackup on tape (OS backups run by SA)”. _______

8.3.5.1 Install system procedures and tables etc in sybsystemprocs

```
cd $SYBASE/ASE-12.5/scripts
isql -Usa -P<> -S<server> <installmaster >installmaster.<server>.<date>
isql -Usa -P<> -S<server> <instmsgs.ebf >instmsgs.<server>.<date>
isql -Usa -P<> -S<server> <installjconnect >installjconnect.<server>.<date>
isql -Usa -P<> -S<server> <installdbccdb >installdbccdb.<server>.<date>
```

____

8.3.5.2 Add objects relating to sybsyntax database
cd $SYBASE/ASE-12.5/scripts

isql –Usa –P<> -S<server> <ins_syn_sql >ins_syn_sql.<server>.<date> _______

8.3.5.3 Install customized procedures in sybsystemprocs

```
cd /home/sybase/sybdba/sql
isql -Usa -P<> -S<server> <sp_thresholdaction.proc
isql -Usa -P<> -S<server> <sp_gpb_displayroles.proc
isql -Usa -P<> -S<server> <sp_gpb_helpsegment.proc
```

____

8.3.5.4 Verify the database looks ok (404 objects for 12.5.0.3) _______

8.3.5.5 Make a full backup of the sybsystemprocs database

```
/home/sybase/sybdba/script/full_backup.csh <server> sybsystemprocs
```
____

8.3.6 <u>Contingency 3: Recovering sybsystemprocs database backups from tape</u>

Estimated time to implement: 1 hr

8.3.6.1 The sybsystemprocs backup file gets backed up onto the OS backup tapes nightly. Use procedure in section 9.2 below, “Restoring a file from ufsbackup on tape (OS backups run by SA)”to load backup file from tape to disk _______

8.3.6.2 Continue with recovery per section 8.3.1 above. Adjust the path name to the backup file _______

8.4 Recovering the sybsystemdb Database

The sybsystemdb database has not been customized for the GPB installation so recovery is very simple.

8.4.1 Using Regular Disk Backup

Estimated time to implement: 5-10 min

8.4.1.1 Load the backup. You may use the science_prod backup for recovery on science_pc and vice versa. The backup generation should also not be important.

```
isql –U<user> –S<server>
load database sybsystemdb from
“/home/sybase/sybsystem/<server>/sybsystemprocs/gen_0/sybsystemdb.full.<date>”
go
online database sybsystemdb
go
```

____

8.4.1.2 Verify sybsystemdb looks ok. _______

8.4.1.3 Backup sybsystemdb database

```
/home/sybase/sybdba/script/full_backup.csh <server> sybsystemdb
```

____

8.4.2 Contingency 1: Backup a working sybsystemdb database and use for recovery

Estimated time to implement: 5-20 min.

8.4.2.1 Dump the sybsystemdb database from a working Sybase server, **not** the one you are working on

```
/home/sybase/sybdba/script/full_backup.csh <working_server> sybsystemdb
```

or, if the regular backup disk is not available

```
isql –Usa –P<pwd> –S<server>
dump database sybsystemdb to “filename”
```

____

8.4.2.2 Continue with recovery per section 8.4.1 above. Adjust the path name to the backup file _______

8.4.3 Contingency 2: Rebuild the sybsystemdb database from scratch

Estimated time to implement: 10-15 min.

8.4.3.1 Backup master and sybsystemdb databases to a separate area so you can try to go back if you have problems. Do not use the regular file structure for these backups.

```
isql –Usa –P<password> –S<science>
dump database master to “<filename>”
dump database sybsystemdb to “<filename>”
```

8.4.3.2 Make an OS-backup of the master device so you can start all over if things get clobbered. The name of the master device is found in the –d parameter in the appropriate RUN_server-file. You’d reverse the input and output files (if and of) during a restore

```
ssh –l sybase <host where server is running>  # sci-base/sci-crunch______
dd if=<device name>  of=<filename> bs=1024k
example
dd if=/dev/vx/rdsk/vol01_01 of=/home/sybase/mstr.sci.dd bs=1024k
```

____

8.4.3.3 Turn off transaction coordination temporarily and recycle the server for the param to take effect

```
ssh –l sybase <host where the Sybase server is running>
isql –Usa –P<password> –S<server>
sp_configure ‘enable xact coordination’, 0
shutdown
cd $SYBASE/ASE-12_5/install
./startserver –fRUN_<server>
```

8.4.3.4 Drop the failing sybsystemdb database. Due to the nature of the database it must be renamed before it can be dropped

```
use master
go
sp_dboption sybsystemdb, “single”, true
```

```
go
use sybsystemdb
go
checkpoint
go
use master
go
sp_renamedb sybsystemdb, faileddb
go
drop database faileddb
go
```

8.4.3.5 Create a new sybsystemdb database using the allocations as noted in the script /sybackup1/carin/sybbackups/<server>/master/gen_0/create_db_for_load.<server>.csh. If the file is missing, use the one in the /home/sybase/sybsystem directory or recover from the OS-backup per section 9.2 below, "Restoring a file from ufsbackup on tape (OS backups run by SA)".

```
create database sybsystemdb on master=4
```
____

8.4.3.6 Verify that the dbid for sybsystemdb is 31513. If that is not the case you may have severe problems if you ever need to rebuild the master device and recover the master database from backups.

```
isql –Usa –P<password> -S<server>
select * from master..sysdatabases where name = 'sybsystemdb'
```
____

8.4.3.7 If the dbid for sybsystemprocs is NOT 31513, make a backup of the database that is 31513, drop it and recreate it once sybsystemdb has been fully recovered.

____

8.4.3.8 Install the syscoordinations table

```
isql –Usa –P<password> -S<server>
use sybsystemdb
go
sp_create_syscoordinations
go
```
____

8.4.3.9 Turn transaction coordination back on and recycle the server for the configuration to take effect

```
ssh –l sybase <host where the Sybase server is running>
isql –Usa –P<password> -S<server>
sp_configure 'enable xact coordination', 1
shutdown
cd $SYBASE/ASE-12_5/install
./startserver –fRUN_<server>
```
____

8.4.3.10 Make backups of both the master and sybsystemdb databases

```
/home/sybase/sybdba/script/save_master_info.csh <server>
/home/sybase/sybdba/script/full_backup.csh <server> sybsystemdb
```
____

8.4.4 <u>Contingency 3: Recovering sybsystemdb database backups from tape</u>

Estimated time to implement: 1 hr

8.4.4.1 The sybsystemdb backup file gets backed up onto the OS backup tapes nightly. Use procedure in section 9.2 below, "Restoring a file from ufsbackup on tape (OS backups run by SA)"to load backup file from tape to disk _______

8.4.4.2 Continue with recovery per section 8.4.1 above. Adjust the path name to the backup file _______

8.5 Recovering the sybsyntax Database

The sybsyntax database should be located on the sysprocsdev device, not on the master device. Do not use the installation script supplied by Sybase but rather a customized script. (The original script has been renamed to ins_syn_sql.orig and the customized script is named ins_syn_sql).

8.5.1 Using Regular Disk Backup

Estimated time to implement: 5-10 min

8.5.1.1 Load the backup. Since the database has not been customized for the GPB installation, you can use the science_prod backup for recovery on science_pc and vice versa. The generation of the backup file on disk is also irrelevant

```
isql –U<user> –S<server>
load database sybsyntax from
 "/home/sybase/sybsystem/<server>/sybsyntax/gen_0/sybsyntax.full.<date>"
go
online database sybsyntax
go                                                                     _______
```

8.5.1.2 Verify sybsyntax looks ok by issuing a command from within ANY database

```
sp_syntax "create catabase"                                            _______
```

8.5.1.3 Backup sybsyntax database

```
/home/sybase/sybdba/script/full_backup.csh <server> sybsyntax          _______
```

8.5.2 Contingency 1: Backup a working sybsyntax database and use for recovery

Estimated time to implement: 5-10 min.

8.5.2.1 Dump the sybsyntax database from a working Sybase server, **not** the one you are working on

```
/home/sybase/sybdba/script/full_backup.csh <server> sybsyntax
```

or, if the regular backup disk is not available

```
isql –Usa –P<pwd> –S<server>
dump database model to "filename"                                      _______
```

8.5.2.2 Continue with recovery per section 8.2.1 above. Adjust the path name to the backup file _______

8.5.3 Contingency 2: Install the sybsyntax database from scratch

Estimated time to implement: 5-10 min.

8.5.3.1 Run the installation script. If the database does not exist at this point it will be created on the sysprocsdev device

```
cd $SYBASE/ASE-12.5/scripts
isql –Usa –P<> -S<server> <ins_syn_sql  >ins_syn_sql.<server>.<date>   _______
```

8.5.3.2 Verify sybsyntax looks ok by issuing a command from within ANY database

```
sp_syntax "create catabase"                                            _______
```

8.5.3.3 Backup master and sybsyntax databases

```
/home/sybase/sybdba/script/save_master_info.csh <server>
/home/sybase/sybdba/script/full_backup.csh <server> sybsyntax          _______
```

8.5.4 Contingency 3: Recovering sybsyntax database backups from tape

Estimated time to implement: 1 hr

8.5.4.1 The sybsyntax backup file gets backed up onto the OS backup tapes nightly. Use procedure in section 9.2 below, "Restoring a file from ufsbackup on tape (OS backups run by SA)" to load backup file from tape to disk _______

8.5.4.2 Continue with recovery per section 8.5.1 above. Adjust the path name to the backup file _______

8.6 Recovering the dbccdb Database

The dbccdb database is used for running dbcc checkstorage. It may not have been installed on all servers and, when installed, may differ significantly. Dbcc checkstorage is typically only run on the standby server. Because of its size, the backup file is located on the designated backup disk, /sybackup1/carin/sybbackups, and not in the sybase home-directory. There is thus no tape backup of this database but losing it is not critical, it can easily be rebuilt.

8.6.1 Using Regular Disk Backup

Estimated time to implement: 5-10 min

8.6.1.1 Load the backup. isql –U<user> -S<server>

```
load database dbccdb from
 "/sybackup1/carin/sybbackups/<server>/dbccdb/gen_0/dbccdb.full.<date>"
go
online database dbccdb
go                                                                 _______
```

8.6.1.2 Verify dbccdb looks ok by issuing a command

```
dbcc checkstorage (orbit_determ)                                   _______
```

8.6.1.3 Backup the dbccdb database

```
/home/sybase/sybdba/script/full_backup.csh <server> dbccdb         _______
```

8.6.2 Contingency 1: Install the dbccdb database from scratch

Estimated time to implement: 5-10 min.

8.6.2.1 Manually run all the steps for creating the dbccdb that have been documented in /home/sybase/sybdba/ddl/dbccdb.create. _______

8.6.2.2 Backup the dbccdb database

```
/home/sybase/sybdba/script/full_backup.csh <server> dbccdb         _______
```

8.7 Recovering the sybdba Database

Although the sybdba database is technically not a system database, for backup and recovery purposes it is treated like one in the sense that it is backed up to the /home/sybase/sybsystem directory weekly. It is not copied to the archive tape and it is also not preloaded to the standby server.

8.7.1 Using Regular Disk Backup

Estimated time to implement: 5-10 min

8.7.1.1 Load the backup. The database will probably only be installed on the science_prod server

```
isql –U<user> –S<server>
load database sybsyntax from
 "/home/sybase/sybsystem/science_prod/sybdba/gen_0/sybdba.full.<date>"
go
online database sybdba
go
```
____

8.7.1.2 Verify sybdba looks ok by retrieving rows from it

```
select * from sybdba..spacestats
```
____

8.7.1.3 Backup sybdba database

```
/home/sybase/sybdba/script/full_backup.csh <server> sybdba
```
____

8.7.2 Contingency 2: Install the sybdba database from scratch

All data will be lost when the database is installed from scratch.

Estimated time to implement: 5-10 min.

8.7.2.1 Create the database

```
isql –Usa –P<> –S<server>
create database sybdba on sybprocsdev
```
____

8.7.2.2 Install the spacestats table per the script in /home/sybase/sybdba/ddl/sp_gpb_helpsegment_____

8.7.2.3 Verify sybdba is working properly by running a proc that inserts data into the spacestats table. The current space stats should be mailed to you immediately

```
/home/sybase/sybdba/script/space_stats <server> archive_print
```
____

8.7.2.4 Backup the sybdba database

```
/home/sybase/sybdba/script/full_backup.csh <server> sybdba
```
____

8.7.3 Contingency 3: Recovering sybdba database backups from tape

Estimated time to implement: 1 hr

8.7.3.1 The sybdba backup file gets backed up onto the OS backup tapes nightly. Use procedure in section 9.2 below, "Restoring a file from ufsbackup on tape (OS backups run by SA)" to load backup file from tape to disk _______

8.7.3.2 Continue with recovery per section 8.7.1 above. Adjust the path name to the backup file _______

9 Miscellaneous Tasks

9.1 Dropping & Recreating a Damaged Database

In some extreme cases a database must be dropped and recreated before it can be recovered from backup. This adds a considerable amount of time to the recovery process and should only be done when necessary. If the physical disk where the database resides is damaged or if a previous drop database command was incomplete a regular drop database may not work. You must then drop it with the dbcc dbrepair command.

Estimated time to implement: depends on the size of the database to drop and recreate.

Drop database:	2.5-5gb/min
Create database for load:	7-14 gb/min
Create/alter database:	2 gb/min

9.1.1 Verify the commands for recreating the database can be found in script _______

```
/sybackup1/carin/sybbackups/<server>/master/gen_0/ create_db_for_load.<server>.csh
Or /home/sybase/sybsystem/<server>/master/gen_0/ create_db_for_load.<server>.csh
```

9.1.2 Drop the database using drop database <database> _______

9.1.3 Create the database. Use the for load option if recovering the database from backup _______

9.1.4 <u>Contingency for missing create-db scripts:</u>

```
execute /home/sybase/sybdba/script/gen_create_db_for_load.csh
use the script /home/sybase/sybdba/wrk/create_db_for_load.<server>.sql
```

____

9.1.5 <u>Contingency for missing disk init scripts:</u>

```
execute /home/sybase/sybdba/script/gen_disk_init.csh
use the script /home/sybase/sybdba/wrk/disk_init.<server>.sql
```

____

9.1.6 <u>Contingency for dropping the database if drop database doesn’t work</u>

Drop database fails, the db still exists in some system tables but not others

9.1.6.1 Check if the database has been marked suspect

- Log in as sa
- Check if the db is marked suspect

```
select name from master..sysdatabases where status & 320 = 320
```

____

9.1.6.2 Mark the database suspect if it is not already

- `use master`
- `sp_configure "allow updates", 1`
- `begin transaction`

```
  update sysdatabases set status = 320 where name = <database>
```

- Verify that only one row was affected and commit the transaction:
  `commit tran`
- `sp_configure "allow updates", 0` _______

9.1.6.3 Recycle the server

/apps/licensed/sybase_local/etc/sybase_stop <server>
/apps/licensed/sybase_local/etc/sybase_start <server> _______

9.1.6.4 Drop the database

```
dbcc dbrepair (<database>, dropdb)
```
____

9.2 Restoring a file from ufsbackup on tape (OS backups run by SA)

Copies of transaction log dumps and system databases are put in the sybase home-directory daily. This and other UNIX directories and file systems are backed up to tape every night Monday thru Thursday by the SA using the UNIX utility ufsdump. These tapes are recycled every week. In addition, weekly incrementals are made on Fridays, and recycled monthly, as well as full backups that are kept indefinitely. The following describes how to recover a file from the ufsdump backup tape to disk. The backed up dump file must be recovered from the backup tape with the ufsrestore command before it can be loaded using the normal Sybase recovery procedures.

9.2.1 Locate the appropriate OS backup tape. The Mon-Friday nightly tapes are stored on top of the science tape drive, ask the SA for other tapes if there is a chance your file resides on a weekly or monthly incremental tape

____

9.2.2 Write-protect the backup tape (flip the tab on the tape to the safe-position) and load it _______

9.2.3 Log in to a science client and go to the /home directory if you want to restore the files to the original directory structure. Please note that existing files will be overwritten. This might be a problem when recovering files without timestamp-extensions to the master-backup directory. If you want to restore to a different directory go there instead. The files will be restored under sybase/syblogs/ or sybase/sybsystem in that directory.

```
ssh –l sybase science
cd /home
```
____

9.2.4 Start the recovery session. Always start each recovery attempt by repositioning the tape to the table of contents in the beginning of the tape

```
/usr/sbin/ufsrestore –t zyxx    ## zyxx is not on volume, that’s ok______
```

9.2.5 Position the tape to the file system that contains the sybase home directory (skip 6). It is important to use the **non**-rewind device

```
/usr/sbin/ufsrestore –ifs /dev/rmt/0n 6
```
____

9.2.6 Locate the files you want to recover by using the commands pwd, cd and ls. The command “add” adds files to the extraction list, “marked” lists the extraction lists for the current directory. You may recover from as many different directories as you wish, just jump around add add files as you go. For instance:

```
ufsrestore > pwd
/
ufsrestore > ls
.:
 TT_DB/      farley/    etc…   sybase/     etc..
ufsrestore > cd sybase/syblogs/science_prod/gpb_3_4_3/gen_0
ufsrestore > ls
./sybase/syblogs/science_prod/gpb_3_4_3/gen_0:
gpb_3_4_3.log.20040710093510   gpb_3_4_3.log.20040711093541
gpb_3_4_3.log.20040710100649   gpb_3_4_3.log.20040711100513
gpb_3_4_3.log.20040710103527   gpb_3_4_3.log.20040711103505
gpb_3_4_3.log.20040710110504   gpb_3_4_3.log.20040711110513
ufsrestore > add gpb_3_4_3.log.2004071010*
#just a warning that you'll be extracting files into an existing directory
structure
ufsrestore > marked
./sybase/syblogs/science_prod/gpb_3_4_3/gen_0:
*gpb_3_4_3.log.20040710100649  *gpb_3_4_3.log.20040710103527
ufsrestore > cd /sybase/sybsystem/science_pc/master/gen_0/
ufsrestore > add master.full.20040712173400
```

```
/master/gen_0:
*master.full.20040712011506
```

```
List the directories/files to extract:
__________________________________________________________________
__________________________________________________________________
__________________________________________________________________
__________________________________________________________________
__________________________________________________________________
__________________________________________________________________
```

____

9.2.7 Start extracting the files from tape and restoring to disk. A sample session may look like this:

```
ufsrestore > verbose
verbose mode on
ufsrestore > setmodes n
Set directory mode, owner, and times.
set owner/mode for '.'? [yn] n
Directories already exist, set modes anyway? [yn] n
ufsrestore > extract
Extract requested files
You have not read any volumes yet.
Unless you know which volume your file(s) are on you should start
with the last volume and work towards the first.
Specify next volume #: 1
extract file
./sybase/syblogs/science_prod/gpb_3_4_3/gen_0/gpb_3_4_3.log.20040710100649
extract file
./sybase/syblogs/science_prod/gpb_3_4_3/gen_0/gpb_3_4_3.log.200407101013527
extract file
./sybase/sybsystem/science_pc/master/gen_0/master.full.20040712173400
Add links
<in some cases you may be prompted for a second volume, escape out>
ufsrestore > quit
```

____

9.2.8 Verify that the proper files were recovered to disk ________

9.2.9 Resume loading the dumps using the regular recovery procedures above

____