12.0 Backup Strategy for FEMIS

Backups are critical in the maintenance of your FEMIS UNIX server since they provide a safety net to prevent data loss in the event of disk failures, software problems, or operator error. Failure to properly backup your system can cause hours or days of unnecessary labor in reproducing lost files and configurations. The ideal backup strategy automates as much as possible, thus minimizing manual actions performed by the System Administrator. However, an improperly implemented strategy can cause problems rather than protect data. If the recommendations outlined below need modifications for your system, please analyze the changes carefully to avoid problems.

This document provides a recommended backup strategy for the FEMIS system and supplies details on using scripts that are installed on the UNIX servers to automate the process and a procedure for implementing system backups on a Sun Solaris system.

12.1 Recommended Backup Strategy

Regularly scheduled file system and Oracle database backups are recommended in addition to manual backups done as part of system upgrades or planned hardware and software maintenance. The backup process should be automated to make sure it always gets done consistently. The best time to backup your system is during times of low use (usually during the night). A full file system backup followed by incremental backups (changed files) is recommended. This will ensure the system can be quickly restored with only a few tapes. A method of tracking taped backups and retention of the media will ensure your ability to recover from data loss.

Some of the data in the FEMIS Oracle database tends to accumulate and can lower performance if it is not periodically removed. The addition of folders in FEMIS takes care of removing old data if folders are used correctly. Scripts are available to save the contents of the database and then remove the folder data that is no longer of use to the operational system.

The Oracle database backups and folder deletion need to be coordinated with the file system backups. This ensures the saved database files are not in the process of being modified while they are being copied to tape, and old database files that are no longer needed on the disk can be removed after a successful tape image is made. If this old data is not removed, the disk can fill up in one to three weeks.

12.1.1 File System Backups

An automated strategy of running full file system backups once a week followed by incremental file system backups the other workdays is recommended. These file system backups must follow the database backups that occur the same night. After a successful full file system backup, the old Oracle export and log files created by the database can be removed.

This process should be repeated each week with different media. For example, at PNNL, we retain 6 months (26 weeks) of full backups and 2 months (8 weeks) of incremental backups. The tapes are numbered and designated as full or incremental backups and kept in numerical order in a cabinet. A logbook is also used to track when tapes were used. Your System Administrator mounts the backup tape each night and then checks the next morning to ensure the backup ran successfully. If a failure of the media occurred, they can then rerun the backup manually. For disaster recovery, the latest full and incremental backups are kept in a different building. This backup regimen has proven to be highly successful in providing us with an efficient way to recover from data loss.

When the FEMIS software was installed on your UNIX server, files system backup scripts and template files were installed and are located in the `install/backup_template` directory. These scripts enable you to schedule and backup your file system. See Section 12.1.2, File System Backup Procedures for the UNIX Server, to customize and setup the server for automated backups. These files contain scripts that will check the full file system backup log for errors before removing the old Oracle export files. This prevents deleting these files without first successfully backing them up.

12.1.1.1 Full File System Backups

A full file system backup creates an image of your system and can be used to restore a disk to the point in time this backup occurred. The operating system tracks the occurrence of a full file system backup of each disk in the `/etc/dumpdates` file on your system unless a third party backup mechanism is used which maintains its own database of backup dates (such as Legato's Networker, AKA Solstice Backup). A full file system backup of a device is designated as a level `0` dump followed by the date and time it occurred, for example:

```
/dev/rdsk/c0t0d0s0  0 Sun Apr 12 00:00:52 1998
/dev/rdsk/c0t0d0s5  0 Sun Apr 12 00:06:04 1998
/dev/rdsk/c0t0d0s6  0 Sun Apr 12 00:11:22 1998
/dev/rdsk/c0t1d0s7  0 Sun Apr 12 00:33:28 1998
```

12.1.1.2 Incremental File System Backups

An incremental file system backup uses the data in the `/etc/dumpdates` file to determine which files have changed since the previous full file system backup and then writes only the changed files to tape (unless a third party backup solution is used as mentioned above). In order to completely restore a disk or directory, the full file system backup must be restored followed by the latest incremental. Incremental file system backups are designated by a level `9` dump in the `/etc/dumpdates` file.

12.1.2 File System Backup Procedures for the UNIX Server

Software backups and archiving are highly recommended as part of normal system administration operations and management. Example scripts are delivered to perform these tasks. The EOC and System Administrator should become familiar with the examples and make any modifications necessary to comply with their information system policies.

The backup files are located in the `install/backup_template` directory and include the following:

`README.backup`	
`backup.sh` -	The script which performs backups.
`backup.sh.1` -	The `backup.sh` man page.
`backup_system_full` -	The control file template for full backups.
`backup_full_data_file_1` -	The data file template for tape 1 of the full backup.
`backup_full_data_file_2` -	The data file template for tape 2 of the full backup.
`backup_system_inc` -	The control file template for incremental backups.
`backup_inc_data_file_1` -	The data file template for tape 1 of the incremental backup.
`backup_check.sh` -	The script to check for successful backups and call the Oracle export and archive log removal script.

To customize the backup templates for your site, complete the following steps:

1. Create the `/apps/backup` directory.

2. Copy the backup files to `/apps/backup`.

3. Configure the backup templates for the system. Each backup data file will write to one tape. If more than two full or one incremental backup tapes are required, create a new data file and add the new data file to the appropriate control file.

To run an Oracle archive removal script:

1. Uncomment the `backup_check.sh` line in the `backup_system_full` file.

2. Edit the `backup_check.sh` script to verify the `EXPECTED_LOGS` variable is accurate.

3. Modify the `ORACLE_REMOVE` variable to call the Oracle file removal script.

To run an automated backup, load the appropriate number of tapes and add the following to the root crontab:

```
#
#       Backups
#
35 0 * * 2 /apps/backup/backup_system_full > /dev/null 2>&1
30 0 * * 3-6 /apps/backup/backup_system_inc > /dev/null 2>&1
```

To perform backups manually, load the appropriate number of tapes and run the following commands.

Full backup:
(performed Monday evenings)

```
# /apps/backup/backup_system_full &
```

Incremental backups:
(performed Tuesday-Friday evenings)

```
# /apps/backup/backup_system_inc &
```

12.1.3 Oracle Database Backups

The Oracle database contains most of the information that is used throughout FEMIS. The database is a critical part of the system. To ensure the database can be restored in case of hardware malfunctions, software problems, or human error, it must be backed up on a regular basis. Although recovery may be complex depending on the types of damage to the database, it can usually be accomplished if the database was properly backed up.

To provide alternative methods of recovery, we recommend the following Oracle database backups be done.

Full database backups copy all the files that comprise the Oracle database. We recommend both periodic "cold" full database backups as described in Section 12.1.3.1, Cold Full Backups of the Oracle Database, and weekly "hot" full database backups as described in Section 12.1.3.2, Hot Full Backups of the Oracle Database.

Logical Oracle database backups are Oracle database exports. We recommend nightly logical Oracle database backups as described in detail in Section 12.1.3.3, Logical Backups of the Oracle Database.

Full database backups and logical database backups provide different recovery capabilities.

Full database backups are used to restore the Oracle database to any point in time, including the last time the database was operating normally. Note that to recover using a full database backup, Oracle should be operated in archive mode so the archive logs are copied to a save area. To recover to a point in time, the last full backup files are loaded, and then the archive log files are applied until the desired point in time is reached. If archive log files are not available, a cold full database backup can still be used to restore the database to the point when the cold full database backup was made, but changes made after that time cannot be recovered. Recovery using a hot full database backup cannot be accomplished unless all archive logs are available.

Logical Oracle database backups are used to recover to the time when the logical database backup was completed. The Oracle import tool is used to regenerate the database in case of major failures. This type of recovery is useful to restore the database to a past state where the database was known to be good. If the database was damaged in some manner so that it would not start up, then imports would not be possible. In this case, the database would then have to be rebuilt using a complex process available in Oracle's installer, or the database could be restored from the most current set of files produced by a cold backup.

It is essential that the database backups be integrated with the file system backups. When this is done, the Oracle files will be ready to be copied to tape along with other disk files, and disk space will be freed when old files are deleted after the successful file system backup. Your System Administrator should ensure the directory containing the archive logs, and the Oracle backup files are included in the file system backup.

When the FEMIS software was installed on your UNIX server, Oracle database backup scripts and template files were also added and are located in the `~oracle/admin` directory. These scripts will enable you to schedule and automate backups for your Oracle database.

12.1.3.1 Cold Full Backups of the Oracle Database

The database must be shutdown to perform an Oracle cold full database backup. A script to perform a cold backup, named `dbbackup_cold`, is available in the `~oracle/admin` directory. This script shuts down the database, copies the files to a save area indicated by the environment variable `ORACLE_COLD` and then restarts the database. In a multiserver configuration, shutting down the database on one server causes replication failures on remote servers since the remote servers continually try to query for database changes. Although these replication failures are temporary and are usually repaired when the database comes back up, sometimes more serious problems are encountered. Therefore, cold backups are not routinely used in FEMIS and are manually initiated at times when the database is shut down for other reasons. Database shutdowns should be coordinated with other remote servers to avoid complications.

Cold backups are recommended before the installation of a new FEMIS version and whenever the server is shutdown for several hours or more for maintenance. This backup can be used to restore the database to the specific date and time it was done. In addition, archived logs can then be applied to restore the database up to the time of the last archive if all archived logs since the last cold backup are available.

12.1.3.2 Hot Full Backups of the Oracle Database

Oracle hot backups are full backups that are done without shutting down the database. A script to perform a hot backup, named `dbbackup_full`, is available in the `~oracle/admin` directory. This script first does a logical backup (see Section 12.1.3.3, Logical Backups of the Oracle Database) and then checks to see if the database is operating in archive mode. If the database is not in archive mode, a hot backup cannot be performed so the script exits. If the database is in archive mode, each data file is put into backup mode, and then it is copied to a save area indicated by the environment variable `ORACLE_FULL`. After that, the Oracle control file is copied to the same save area. At this point, the database is backed up, and the files in the save area can be copied to tape as part of the file system backup process. When all files are safely backed up to tape, the online Oracle redo logs are removed so the file space is available for the next set of logs.

It is recommended that hot backups be done weekly during off-use time when changes to the database are minimal. These backups can be used with the archive logs to restore the database to a point in time. All database archive logs, from the time the hot backup was started to the time of

desired recovery, must be available in order to restore the database. If logs are missing, the hot backup will not succeed—for this reason, cold backups are considered essential.

12.1.3.3 Logical Backups of the Oracle Database

A logical Oracle database backup uses the Oracle export utility tool to make a consistent copy of the database to a file. Logical Oracle backups are combined with folder deletion to ensure that closed folder data is saved before the delete process removes it. A script to perform folder deletion, named `dbbackup_folder`, is available in the `~oracle/admin` directory. A system level export dumps all Oracle objects in all Oracle user accounts to the save area indicated by the environment variable `ORACLE_EXPORT`. A typical logical backup takes about 15 minutes, and after this time, the export file is ready to be copied to tape by either a full or incremental file system backup. A logical Oracle database backup does not require the Oracle database to be shut down.

It is recommended that logical backups be done each working day during low use times to save the database as it exists. From this export, individual user accounts can be restored using the Oracle import tool. When this is done, data in all tables are restored to what existed at the time of the export.

Also, data in a specified set of tables can be restored from a logical Oracle database backup, leaving the rest of the database alone. This can be useful if data in a table is deleted accidentally because restoration to a previous day's logical Oracle database backup will save time by not having to recreate the lost data.

12.1.4 External Storage of Folders and Deletion of Old Folder Data

As the FEMIS system is used, data accumulates in many of the Oracle tables. Certain tables may get extremely large and slow down the performance of the system. The meteorological, D2PC, and journal log data, all of which have frequent updates, are of special concern. Some EOCs wish to maintain a record of this information for an extended period of time, so some data cannot be simply deleted. Folder processing allows historical data to be saved for possible future use and deletes this information from the operational system.

A more complete description of the database aspects of folders can be found in the Section 9.0, Folder Management and Archiving, in the ***Data Management Guide for FEMIS Version 1.5.3***. A brief description related to backing up the Oracle database followsn. A template is provided in the `/oracle/home/admin` directory for use as a crontab table for the UNIX `femis` user. When this is implemented, the folder deletion process will be done each workday evening. First a system level export is performed with the output file generated in the `ORACLE_EXPORT` directory. If this export completes successfully, the folder delete process then checks to see if folder data can be removed from the database. Normally folder records are then removed.

Since meteorological data is folder independent, it is handled as a special case in the folder delete process. Meteorological data is checked each Monday and any records older than 7 days will be removed. Journal data is a folder table, but it is checked on the first Monday of the month. Any data older than 30 days will be removed.

D2PC cases are saved in folder tables so that data is normally deleted along with the other folder records. In certain cases, D2PC may accumulate over time so a script is available to manually remove this data. This process can be configured to operate automatically as a cron job, or it can be used interactively. It is recommended that the archiving of D2PC cases be tailored to your EOC and configured to operate automatically if D2PC case buildup is a concern.

12.1.5 Managing the FEMIS Log Files

As the FEMIS system is used, log files are created and accumulate. In particular, the FEMIS Notification Service and Command Server generate log files daily. The `manlog` option of the database backup program executes the `manage_femis_logs.sh` script that removes log files older than two weeks and bigger than one megabyte in size. During installation, the FEMIS crontab is setup to execute this program daily at 2:13 am to ensure that the files are removed on a regular basis. If managing these log files manually is desirable, then simply remove or comment out the entry in the FEMIS crontab file.

12.2 System Backups for Sun Solaris System

The following is a procedure for implementing system backups on a Sun Solaris system using the PNNL developed `backup.sh` script and data files.

1. Create a directory on your Sun server to keep your backup logs and scripts. A commonly used location is `/filesystem/apps/backup`. You can add an entry to your `/etc/auto_apps` to automap this directory as `/apps/backup`.

   ```
   # apps directory map for automounter
   #
   backup   -intr,rw,nosuid   system:/files0/apps:&
   ```

2. Copy all files located in `/home/femis/install/backup_template` to your backup directory.

3. Document your system's configuration for the following items:

   - Number of bytes that your tapes are able to store on your tape drive.
   - Tape drive device address (e.g., `/dev/rmt/#`). If it is the only tape device on your system, it is likely to be `/dev/rmt/0`.

- Appropriate `ufsdump` options for your tape device (see the man pages on ufsdump and tape drive manufacture's specifications).

- Mount point of system disks.

- Disks size and bytes used.

- Document the directory where your oracle home account is located if you are going to remove Oracle exports after your full backup.

4. Configure each of the backup data files to match your system's configuration. Modify, if necessary, the following items:

   - `Options` – This is the `ufsdump` options for your device. The template files are configured for 4mm DDS tape drives. The first option is for dump level and should be left as either `0` for a full backup or `9` for an incremental backup. You need to include the `u` and `f` options regardless of tape drive used.

   - `Device_file` – This is the tape drive device address. If your tape drive can compress data, include the `c` parameter. Always include the `n` parameter (e.g., `/dev/rmt/#cn`).

   - `Filesystem` – This is the mount points of the system disks (space delimited). Your typical incremental backup will include all file systems. Most full backups will need two or more full data files. Do your best to arrange them so tapes do not run out of space. Do not duplicate or leave out any disk drives.

   - `Mail_to` – This is a list of UNIX accounts or E-mail addresses (space delimited), which will receive the backup log and a warning list at the end of the each backup tape.

5. Each backup data file will write to one tape. If you need more than two full or one incremental backup data files, make a copy of an existing file and name it according to the order it will be used. Edit and change the `log_file` option to match the data file number.

6. Add/remove lines in the `backup_system_full` and `backup_system_inc` files so they execute all the data files with the `backup.sh` script. Be sure a `sleep (a minimum of 180 seconds, shipped specified as 360 seconds)` command separates each backup execution for autoloaders. This command gives the tape drive time to unmount and remount the tapes.

7. Uncomment the `backup_check.sh` line in the `backup_system_full` file to run an Oracle archive removal script. You will also need to edit these variables in the `backup_check.sh` script:

   - `ORACLE_REMOVE` – This line will be `oracle_home_directory/admin/dbbackup_cron -clean`.

- `EXPECTED_LOGS` – This will be the number of backup logs generated by the full backup.

- `LOG_PATH` – The directory where these logs are located.

When this script runs, it mails its results to the root mail account by default. The E-mail account can be changed by editing the `backup_check.sh` script. Modify the following section (near the bottom) by replacing `root` with the E- mail account you want to receive the results.

```
if [ -f "$LOG" ];
  then
    < $LOG mailx -s "Oracle Export Removal $REMOVAL_STATUS " root
    rm $LOG
fi
```

8. Load the appropriate number of tapes each night, and add the following to the root crontab to run an automated backup:

```
#
#       Backups
#
35 0 * * 2 /apps/backup/backup_system_full > /dev/null 2>&1
30 0 * * 3-6 /apps/backup/backup_system_inc > /dev/null 2>&1
```

This entry in the root cron will execute a full backup at 12:35 am Tuesdays and incremental backups Wednesday through Saturday at 12:30 am. To perform backups manually, load the appropriate number of tapes and run the following commands as root.

Full backup command: `# /apps/backup/backup_system_full &`

Incremental backup command: `# /apps/backup/backup_system_inc &`

9. Label and date your tapes.

Do not reuse the same tape for each backup. You should keep several good tape backups on hand at all times. Determine how long you want to retain full and incremental backups and purchase sufficient tapes to cover that time. You should also purchase extra tapes to be able to replace bad tapes. Your full backups should be kept significantly longer than incremental backups and keep full backups separate from your incremental backups. Mount the oldest incremental or full tape each time backups run.