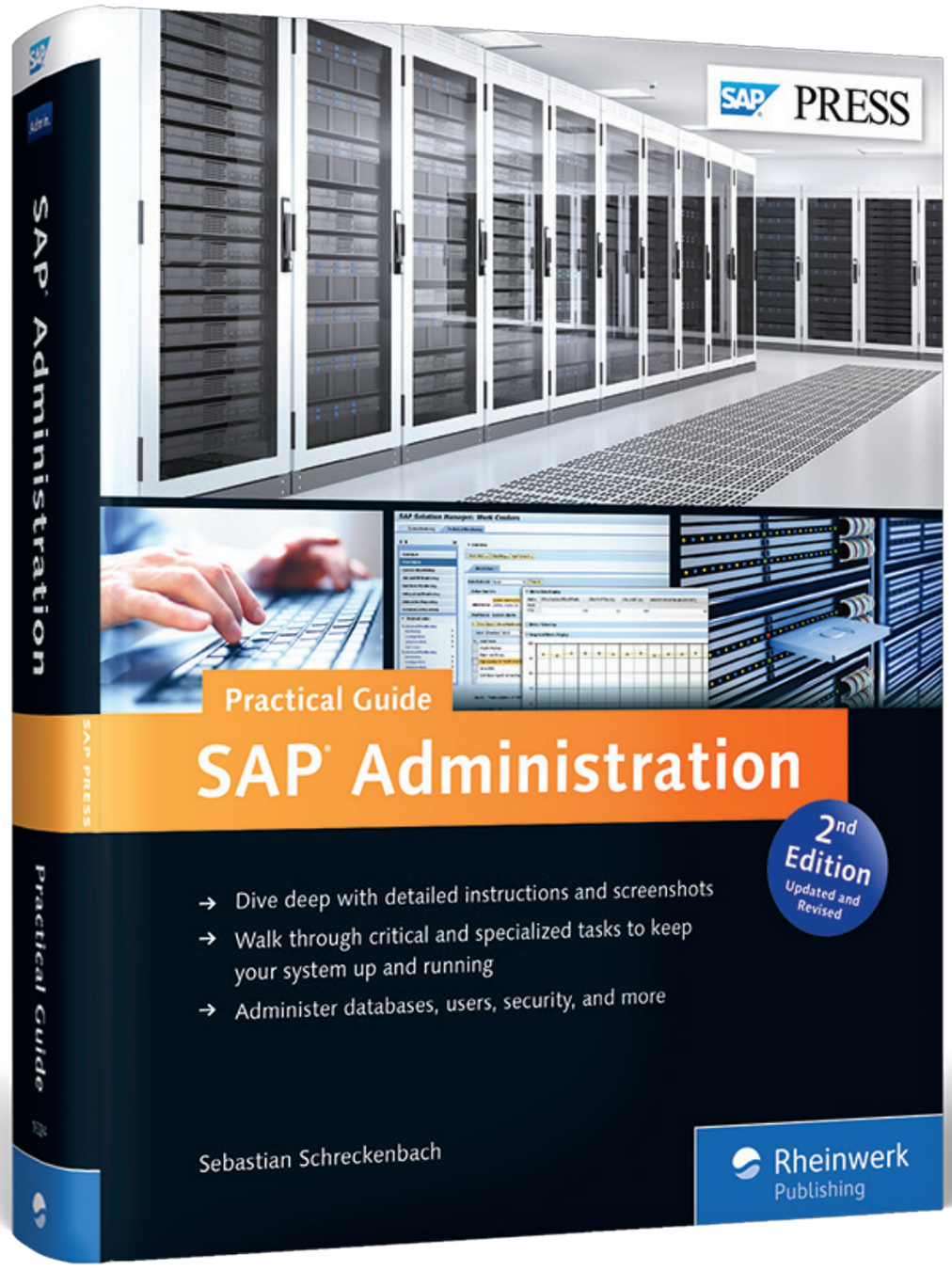


Reading Sample

Preparing for worst-case scenarios is just part of the job for a system administrator. With this reading sample, discover vital backup and recovery strategies for protecting your SAP system during a failure. Then, learn what to do when disaster does strike, and the steps you need to know to recover from a serious malfunction.

Sebastian Schreckenbach
SAP Administration—Practical Guide
912 Pages, 2015, $79.95/€79.95
ISBN 978-1-4932-1024-4
www.sap-press.com/3639

This chapter explains how to develop a backup and restore strategy. After covering the main backup methods, we'll explore the benefits and drawbacks of each. You can then use this information to develop an appropriate backup concept for your own SAP system.

6 Backup and Restore

An effective backup and restore strategy forms the backbone of SAP system operation. The goal of this strategy is to enable a full or partial recovery of the database in as short a time as possible following system failure, an emergency situation (see Chapter 7), or a hardware/software error.

The information provided in this chapter is intended to help you develop a concept to optimize the continuous backup of your data and allow you to restore your database quickly and efficiently in the event of an emergency. We begin by discussing the two aspects of backup and restore, before turning to the issue of performance. Details relating to individual databases are provided in Chapter 8.

The goal of a backup strategy is to minimize data loss in the event of an emergency, in other words, to make sure that no data are lost or to minimize the period during which data are lost. To achieve this objective, your backup strategy should be as clearly defined as possible because an unnecessarily complicated strategy may also make your backup and restore processes unnecessarily complicated. You should also ensure that your procedures and handling of problems are well documented and that your backup strategy doesn't impact negatively on your enterprise's routine business operations.

6.1 Backup

The purpose of a system backup is to allow you to access the data currently stored in the system and to import that data back into the system following an emergency. This is a safeguarding measure because you'll only need to use the backup

if your system needs to be restored, unless you plan, for example, to build a quality assurance system from a system backup. Nevertheless, backups aren't a trivial matter and shouldn't be treated as such. On the contrary, you should take a moment to consider how much data can be lost in the event of a system failure and what ramifications this may have for the enterprise. Even if you lose order data for only an hour or a day, the economic impact on your business may be huge.

6.1.1 What Has to Be Saved?

Three categories of data require backup:

- Databases
- Transaction logs
- Operating system files

You may need to use different tools to back up different data. For example, SAP tools only allow you to back up one or two of the data categories; for example, the DBA Planning Calendar (Transaction DB13; see also Chapter 8) is capable of backing up your database and transaction logs but not your system files.

Databases

The database represents the very heart of your SAP system. Without a backup of your database, you won't be able to restore the system. The frequency with which you perform a full database backup determines how far back in time you must go when restoring the system:

- If a full backup is performed every day, you require the full backup from the previous day, as well as the transaction log files from the last day or last half-day to restore the system.
- If a full backup is performed every week, you require the full backup from the previous week. However, you must also recover the log files from the last number of days to update the system.

A daily backup reduces the risk of your being unable to restore the current database status if you're unable to use the relevant log files.

If you don't perform a daily backup, you require a large number of log files to update the system. This step increases the duration of the restore process due to the volume of files involved and also increases the risk of your being unable to

restore the database to the current status due to individual defective transaction logs.

Weekly Backups [Ex]

A restore is performed using the full backup from the previous week, which dates from four days ago. Keep the following in mind:

- Ten log files are created every day.
- As a result, the system must be updated with 40 files (10 log files × 4 days).
- You require 120 minutes to load the log files from the tape to the hard drive (40 files × 3 minutes per file).
- You require 200 minutes to update the database with the log files (40 files × 5 minutes per file).

The total time required for the restore—not taking account of the actual database files—is 320 minutes (or 5.3 hours).

Daily Backup [Ex]

A restore is performed using the full backup from the night before. Keep the following in mind:

- A maximum of 10 log files are created every day.
- You require 30 minutes to load the log files from the tape to the hard drive (10 files × 3 minutes per file).
- You require 50 minutes to update the database with the log files (10 files × 5 minutes per file).

The total time required for the restore—not taking account of the actual database files—is 80 minutes (or 1.3 hours).

As illustrated in these examples, a restore takes much longer to perform if backups are made on a weekly rather than a daily basis. These examples also demonstrate that the time required to restore the log files depends on the size of the files and on the number of days that have elapsed since the last full backup was performed. The process can quickly become unmanageable in the case of large log files (e.g., 100MB or more per hour). By performing full database backups on a more frequent basis (i.e., by leaving fewer days between backups), you can automatically reduce the time required for a restore.

You therefore need to weigh the question of whether performing a full database backup on a daily basis with fewer files is a more viable option for your system,

based on the volume of transactions involved, than, for example, performing a full backup on a weekly basis with an accordingly higher number of transaction logs to be reloaded. Of course, this decision also depends on the importance of the SAP system in the context of your enterprise's business processes.

[+] **Daily Full Database Backups**

You should have a very good reason not to perform a daily backup of your production database (e.g., your database is too big to be backed up overnight). In recent years, the cost of storage space has fallen to such a degree that a backup strategy based on a full daily backup is no longer impractical. SAP recommends that you perform a fully daily backup of the production database and that you store the 28 most recent backups.

Transaction Logs

Transaction logs form part of a database backup and are essential to performing a database restore. These logs contain all changes made to the database. They allow you to undo these changes and to restore the database to its most recent status after a system failure. It's essential to have a complete backup copy of all transaction logs. If even a single log can't be used when you need to perform a restore, the database can't be restored beyond the point at which this gap occurs.

[Ex] **Damaged Log Files**

A log file from Tuesday is damaged. The system fails two days later, on Thursday. You can only restore the database up to the last error-free log from Tuesday. From the point at which the damaged log occurs, all subsequent transactions are lost.

The frequency of log backups is also a business decision, based on the following factors:

- Transaction volume
- Critical periods for the system
- Volume of data that management can tolerate losing
- Resources required for the backup

[+] **Intervals Between Log Backups**

The following principle applies here: The greater the volume of transactions, the shorter the intervals that you should leave between the individual log backups. In this way, the

volume of data that can be lost in the event of a potential disaster in the data center is automatically reduced.

To back up the transaction logs, follow these steps:

1. Save the transaction log to the hard disk.
2. Copy this backup to a backup file server located at another site. You should always use verifications when you save your logs across a network.

   The backup file server should ideally be located in another building or another city. A remote location increases the chances of your backup remaining intact if the primary data center (containing the SAP servers) is destroyed.
3. Save the transaction log backups from both servers (the SAP server and backup file server), together with the other files from the operating system level to tape on a daily basis.

[!] **Database Stops When the Backup Directory Is Full**

Transaction logs are stored in a directory, which must have sufficient storage space. The database stops when the available memory in the directory is completely occupied by the transaction logs. If no further processing can take place in the database, the entire SAP system stops also. It's therefore important to think ahead and to back up transaction logs on a regular basis.

If a backup file server in a separate location isn't available to you, you must save the transaction log backups to tape after each log backup operation and send the tapes to another location on a regular basis.

[!] **No Backups in Append Mode**

Don't back up the logs to tape in *append mode*. In this mode, several backups are written to the same tape. In the event of an emergency, all backups on this tape may be lost.

Files at the Operating System Level

You also need to back up files at the operating system level:

- Configuration of the operating environment (e.g., system and network configuration)
- SAP files (e.g., kernels)
- System profiles

- Spool files
- Transport files
- Other SAP-related applications
- Interfaces or add-on products that save their data or configurations outside of the SAP database

The data volume of these files is relatively small compared to the SAP database. Depending on how your system works, the backup of the files in the list may only comprise a few hundred megabytes to a few gigabytes. In addition, some of the files may contain static data, which remains unchanged for months at a time.

The frequency of backups at the operating system level depends on the applications involved. If you need to ensure synchronicity between these application files and the SAP system, they must be backed up with the same frequency as the logs.

[Ex] **Synchronizing Application Files and the SAP System**

One example of this scenario is a tax calculation program, which stores VAT data outside of the SAP system. These files must correspond exactly to the sales orders in the system.

A quick and easy method of backing up operating system files is to copy all files to the hard drive of a second server. A range of products for backing up data at the operating system level is available on the market at the present time. You can then back up all required files to tape from the second server. This approach minimizes the periods during which files are unavailable.

6.1.2 Backup Types

We can distinguish between different types of database backups based on the following three questions:

- What is backed up? Is the backup a full or incremental backup?
- How is it backed up? Online or offline?
- When is it backed up? Is the backup scheduled or ad hoc?

You can, in principle, combine the various answers to these questions to produce a range of options. Each variant has its benefits and drawbacks, which are discussed next.

What Is Backed Up?

In terms of the scope of the database backup, you can choose between a full or partial backup, as described here:

- **Full database backup**
  Note the following considerations:
  - *Advantages*: The database as a whole is backed up, which makes a database restore faster and easier to perform. Fewer transaction logs are required to update the database.
  - *Disadvantages*: A full backup takes longer to complete than an incremental backup. As a result, users are disrupted for a longer period. You should therefore only perform full backups outside of normal business hours.
- **Incremental backup with transaction logs**
  Note the following considerations:
  - *Advantages*: An incremental backup is much quicker than a full database backup. Because the backup takes less time to complete, users are impacted for shorter periods and, in most cases, to a barely noticeable degree.
  - *Disadvantages*: A full backup is required to restore the database. Restoring the database with incremental transaction logs takes much longer and is more complicated than a restore based on a full backup. The most recent full backup must be used for the restore, and the system then has to be updated with all logs dating from the time when the full backup was made. If several days have elapsed since the last full backup, a very large number of logs have to be restored if the system fails. If you're unable to restore one of these logs, you'll also be unable to restore any subsequent log.

A third option may also be available to you, depending on your database and operating system (see Table 6.1):

- **Differential backup**
  In this case, you only back up the changes that have been made since the most recent full backup. One commonly used approach is to perform a full backup every weekend and differential backups during the week.
  - *Advantages*: The risk of your being unable to perform a full restore because of damaged log backups is reduced. A differential backup saves all changes made to the database since the last full backup.

- *Disadvantages*: As with an incremental backup, you still require a full backup as a basis for restoring the database. A differential backup may take longer to complete than a backup of the transaction logs. Initially (after the full backup), it will take less time, but the process will gradually become longer over time as more data are changed.

[!] **Full Backup as a Basis for an Incremental Backup**

Note that an incremental backup always comprises a full backup *and* a backup of the subsequent transaction logs. A restore based on incremental backups becomes problematic as soon as the underlying full backup or one of the transaction logs is damaged or lost.

How Is It Backed Up?

In terms of backup mode, we can distinguish between *offline* and *online*, based on the system status of the SAP system and the database. To perform an offline backup, you must disconnect the SAP system from the database and stop work in the SAP system. An online backup, on the other hand, is performed during normal operation of the database and SAP system.

- **Offline**

  Benefits

  - An offline backup is faster than an online backup.
  - There are no complications caused by changes to data in the database during the backup.
  - All files are backed up at the same time and give a consistent picture of the system; the corresponding operating system files are synchronized with the SAP database.
  - You can execute a binary verification during an offline backup. However, this doubles the time required to perform the backup.
  - An offline backup doesn't require the SAP system to be stopped. The SAP buffer is therefore preserved.

  Drawbacks

  - The SAP system isn't available during an offline backup.
  - When the database is stopped, the database buffer is also cleared of all data. This operation has a negative impact on performance, with this effect lasting until the buffer is filled with data once again.

- **Online**

  Benefits

  - The SAP system is available to users during the backup. This is essential if the system is in constant demand 24/7.
  - The buffers aren't cleared. As a result, there is no negative impact on performance following the backup.

  Drawbacks

  - An online backup is slower than an offline backup. The time taken to complete the backup increases over time because the backup runs during normal operation and uses system resources.
  - Online performance deteriorates during the backup.
  - The data in the database may change while the backup is still in progress. Transaction logs are therefore particularly important to ensure a successful restore.
  - The corresponding files at the operating system level may possibly no longer be synchronized with the SAP database.

Transaction Logs in an Online Backup [!]

If you use online backups, transaction logs are particularly important to ensure a successful restore.

When Is It Backed Up?

You can select the time at which a database backup is performed based on a backup schedule or spontaneously as the need arises in a specific situation. For more information about the tools and transactions mentioned in the following, refer to Chapter 8:

- **Planned**

  Planned backups are performed on a regular basis, for example, daily or weekly. For normal operation, you can use the DBA Planning Calendar (Transaction DB13) to configure an automated backup schedule for the database and transaction logs. You can use this calendar to set up and check backup cycles. You also have the option of performing important database checks and updating the statistics. You can display the status of your backups in the DB Backup Monitor (Transaction DB12).

- **Ad hoc**
  Ad hoc backups are spontaneous backups performed on an as-needed basis, for example, prior to large-scale system changes, in preparation for an SAP upgrade, or after a structural change to the database (such as the addition of a data file). Backups that are monitored directly by the user or are performed on an as-needed basis can either be initiated using the DBA Planning Calendar or at the database or operating system level.

The DBA Scheduling Calendar can be used for both regular, planned backups and spontaneous backups. However, tools at the database level, such as SQL Server Management Studio for Microsoft SQL Server or BR* Tools under Oracle, are more commonly used for these ad hoc backups. Regardless of the backup method you select, you should always set the following goals:

- Create a reliable backup that can be used to restore the database.
- Use a simple backup strategy.
- Reduce the number of interdependencies required for operation.
- Try to eliminate or minimize the impact on the work being done in the system by business department users.

Weigh up the needs of system security and performance to use the available options to develop the best possible backup strategy for your system.

Database System-Specific Terminology

Table 6.1 compares the terminology that is used in relation to the various methods outlined in the previous sections and for backing up various database systems. The backup methods and jobs have different names, depending on which database your system uses. However, the underlying principle is always the same. If in doubt, consult your database administrator or the documentation provided for your database system.

	Full Database Backup	Content	Partial Database Backup	Log Backup
DB2 UDB	Full database backup in TSM (Tivoli Storage Manager)	Offline/online tablespace backup in TSM	Incremental database backup with DB2 UDB in TSM	Archiving of inactive log files in TSM

Table 6.1 Terminology of Backups

	Full Database Backup	Content	Partial Database Backup	Log Backup
	Full database backup to storage device	Offline/online tablespace backup to storage device	Incremental database backup with DB2 UDB to storage device	Archiving of inactive log files on storage device
	Full database backup with vendor library	Offline tablespace backup with vendor library	Incremental database backup with DB2 UDB and vendor library	One-step archiving in storage software
SQL Server	Full database backup		Differential database backup	Transaction log backup
Oracle	Full database backup offline and new log backup	Full offline database backup	Partial offline database backup	New log backup
	Full database backup online and new log backup	Full online database backup	Partial online database backup	

Table 6.1 Terminology of Backups (Cont.)

6.1.3 Backup Strategy

Your backup strategy unites and defines all measures used to back up your system and specifies when exactly backups are to be performed, the intervals at which they are to be performed, and the backup method that is to be employed. You should document this strategy in the form of a *backup frequency table* in a backup concept and ensure that it meets the needs of management and the business departments.

You then implement your *backup strategy* with the appropriate backup tools. Ultimately, however, your choice of tool to implement the strategy is of little relevance, be it one of the SAP-internal tools mentioned previously or the standard tools provided in your database or operating system. The most important criteria when selecting tools are manageability, reliability, and the monitoring options.

To develop a backup strategy, follow these steps:

1. **Determine your requirements for performing a restore and your tolerance range in the event of a system failure.**
   A generally acceptable system downtime can't be defined because this will differ significantly between one enterprise and the next. The costs incurred by system downtime include the cost of production downtime, plus the costs of performing a restore, such as time, money, and so on. These costs should have a sliding scale, similar to that used for insurance premiums. With insurance, the more coverage you require, the greater the premium you have to pay. If we apply this model to a system restore, we get the following rule: The faster a restore is completed, the more expensive the solution you'll have to use.
2. **Determine which combination of hardware, software, and processes is used in the desired solution.**
   Better hardware makes a backup and restore faster, better software makes these operations easier, and well-defined processes make them more efficient. Of course, this all comes at a price, and the benefits will have to be weighed against the costs. However, it's even more important that your method be reliable.
3. **To test your backup method, implement the hardware, and check the actual runtimes and test results.**
   Ensure that you obtain results for all backup types used in your environment and not only those you intend to use. This information will facilitate future evaluation and capacity planning decisions and, if necessary, provide a sound basis for comparison.
4. **Test your restore method by simulating various system failure scenarios.**
   Document all aspects of the restore; include questions such as who will take care of specific tasks, which users are to be notified, and so on (see Chapter 7). You should also consider the likelihood that a restore may occur exactly when you least expect it. You should therefore conduct testing on an ongoing basis and perform additional tests whenever changes are made to hardware or software components.

Schedule additional backups on specific dates (e.g., end of the month, end of the year) alongside your daily and weekly backup cycles. These aren't strictly necessary but can, for example, be archived separately as a safeguard against a disaster (see Chapter 7).

6.1.4 Strategy Recommendations

This section provides some further tips and recommendations for developing a backup strategy.

Databases

As discussed previously, we recommend that you perform a full database backup every day, provided that the cost of doing so isn't prohibitively high. If your database is too large for a daily backup, you should perform a full backup once a week instead.

Testing Your Backups

Your backups need to be tested on a regular basis. To do this, you need to restore the system and then conduct a test to determine whether the restore has been completed to your satisfaction. Without testing your backups, you can't tell whether all of the required data has actually been backed up on the tape or hard disk.

Why Testing Your Backups Is Essential [Ex]

Various files were backed up, but the APPEND switch was set incorrectly for the second file and all subsequent files. As a result, the files weren't saved to tape in sequence. Instead, the tape was rewound after each file was backed up and prior to the backup of the next file. The outcome is that all files except for the last file to be backed up were overwritten.

Test Finished Backups Only [!]

You can only test a backup after *all* files have been backed up. If you test your backup after each individual file, the system will be unable to detect whether the previous file has been overwritten.

Database Integrity

You need to check the integrity of the database regularly to ensure that it contains no damaged blocks. Otherwise, defective blocks may remain undetected during a backup. If possible, conduct an integrity test once a week outside of business hours. This can be scheduled with the DBA Planning Calendar.

Transaction Logs

It's extremely important that you back up your transaction logs. The database and, therefore, also the SAP system, stops when the memory that is available for storing the transaction logs is full.

For this reason, monitor the number of transaction logs in your system, and define your own backup interval, for example, hourly, based on your monitoring findings. The intervals between the backups correspond to the maximum data volume that you can tolerate losing. The risk is naturally higher for an enterprise with a large transaction volume. In this case, it would be advisable to perform a backup every 30 minutes, for example. If your enterprise has a shipping department that starts work at 3:00am, or a production line that works until 10:00pm, you should begin making backups earlier or stop later as required. Transaction logs can be backed up during normal operation without any impact on users.

Files at the Operating System Level

The frequency of backups at the operating system level depends on the applications involved. If you need to ensure synchronicity between the application files and the SAP system, they must be backed up with the same frequency as the database and logs. If perfect synchronicity is less important, you can also back up the application files less frequently.

Backup Strategy Checklist

You need to develop an appropriate system for backing up valuable system data. You should define a suitable strategy as soon as possible to avoid a possible loss of data. You should have worked through a checklist covering all backup-relevant topics before your system goes live (see Table 6.2).

Question, Task, or Decision	Done
Decide how frequently you want to perform a full database backup.	
Decide whether partial or differential backups are required.	
Decide whether to use automatic backups. If you want to use automatic backups, decide where to do this (in the DBA Planning Calendar or elsewhere).	

Table 6.2 Backup Strategy Checklist

Question, Task, or Decision	Done
Decide how frequently the transaction logs are to be backed up.	
Define which backup media (hard disks, tapes, etc.) you want to use.	
Ensure that you can store a day's volume of logs on the server.	
Ensure that you have sufficient memory in the directory for transaction logs.	
Set up the authorizations required for the SAP system, the operating system, and the database.	
Consider whether you want to use the DBA Planning Calendar to schedule the backup of transaction logs.	
Work out guidelines for labeling data carriers to ensure a smooth workflow.	
Decide on the period for which your backups are to be stored.	
Acquire the required hardware (hard disks) or define the size of the tape pool required (tapes required per day × retention period + 20%).	
Take account of future growth and special requirements.	
Initialize the tapes.	
Define a storage strategy for the tapes.	
Document the backup procedures in an instruction manual.	
Train users in the backup procedures.	
Implement a backup strategy.	
Perform a backup and restore for testing purposes.	
Define a contingency plan for emergencies, and decide which users are to be contacted in the event of an emergency.	

Table 6.2 Backup Strategy Checklist (Cont.)

6.2 Restore

You usually perform a restore for one of the following reasons:

- Disaster recovery following an emergency situation (see Chapter 7)
- Testing of your disaster recovery plan (see Chapter 7)
- Copying your database into another system (see Chapter 2)

You access the backups that are made on a regular basis to perform a system restore. In the context of disaster recovery, you usually restart the database and,

if necessary, the operating system, using the most recent full backup. You then import the transaction logs that have been created since the full backup. When this procedure is successfully completed, the system once again has the status it had at the time the last error-free log backup was made. The duration of this restore is of critical importance. You want it to be completed as quickly as possible so that the system can be used again after an outage, and the disruption to business processes can be kept to a minimum.

For a *database copy* (e.g., in the context of regular updating of the QA system using a copy of the production system), you normally either import the most recent full backup or generate a live copy using data streaming. Transaction logs are usually ignored.

As in your system backup strategy, you should also have a *restore strategy* in your arsenal, which can be deployed in the event of an emergency. The following factors may influence your restore strategy:

- Business costs incurred by system downtimes
- Operational schedules
- Global or local users
- Number of transactions per hour
- Budget

The development of a restore strategy is discussed in detail in Chapter 7. The actual process of restoring the SAP system and database isn't discussed in this book because this task varies widely between different systems and databases. If in doubt, consult an expert (e.g., your database administrator or an external Basis consultant) who can provide you with operational support for this critical process. You should also collaborate with your database administrator or consultant to test and document the restore process. This transfer of knowledge will soon enable you to perform a restore on your own.

[!] **An Incomplete or Incorrect Restore**

If the restore is performed incorrectly or incompletely, it may fail and have to be restarted to avoid the possibility of some files being excluded. Certain data must be entered via your database so that it can be restored subsequently. Work with an expert to identify and document this data.

Because the restore process is one of the most important tasks in the SAP system, you need to test database restores at regular intervals. Chapter 7 provides additional information on this.

6.3 Performance

The key objectives of a database restore are to restore the data as completely as possible and to minimize the time required to do so. The length of time that the SAP system is unavailable to users and, as a result, certain business processes are halted, is of critical importance to an enterprise. System performance is therefore a key factor when performing a restore.

The performance of your backup process is also important, in particular if your system is used globally 24/7. Disruption to users should be kept to a minimum during a backup. As a result, you need to strive to reduce the duration of the backup (in particular, in the case of offline backups) and to ensure adequate system reserves to guarantee acceptable system operation during an online backup.

The performance of your backup and restore processes are largely determined by data throughput on your devices. To improve throughput, you need to identify bottlenecks or devices that are limiting the throughput and eliminate or replace these. This process is subject to economic considerations because performance enhancement with additional or more modern devices is naturally also a cost factor.

This section provides tips for improving the performance of your data backups and restores by implementing some specific measures.

6.3.1 Performance Factors

The main variables, which are provided in the following list, affect the performance of both the backup and the restore:

- **Size of the database**
  The larger the database, the longer it takes to back it up.
- **Hardware throughput**
  This variable determines how quickly the backup can be performed. Throughput is always determined by the weakest link in the backup chain, for example:

- Database driver array
- Input/output channel (I/O) channel used
- Hard disk or tape drive

- **Time of backup**
  This is the time or period available to you for regular system backups. Your objective here should be to minimize disruption to users. Consider both online and offline backups:
  - *Online backup*: The appropriate times for performing online backups are periods during which there is a low level of system activities, which is usually early in the morning.
  - *Offline backup*: The appropriate times for performing offline backups are periods during which you can shut down the SAP system, which is usually at the weekend.

  The times at which you perform system restores are less critical because the system can't run in any case unless you do so.

[+] **Take into Account the Time Differences Between Different Sites**

Remember to take into account the time differences between the various sites in which your enterprise is located. For example, when it's 12:00 midday in Central Europe, it's only 6:00am in New York.

6.3.2 Backup Performance

The following approaches to improving backup performance assume that you save your backup locally on the database server. Although a backup via the network is technically possible, performance in this case depends to a large degree on network topology, overhead, and data traffic, while the throughput values of the disk systems take a backseat. In any case, the full capacity of the network is rarely available. If you perform a backup via the network, network performance also deteriorates for other users. As a result, other applications in your enterprise may be slowed down.

Backup to Faster Devices

All approaches to optimizing performance aim to prevent bottlenecks occurring on the backup device. The backup device, usually a hard disk or tape drive, is the

device that limits throughput. You should consider the following aspects in this context:

- **Advantages**
  Faster hard disks or tape drives allow you to save an entire database within a reasonable amount of time.
- **Disadvantages**
  Fast memory is more expensive. Hard disks or tape devices with high data throughput require willingness to invest.

Parallel Backup

A parallel backup to more than one tape drive uses a RAID-0 array (Redundant Array of Independent Disks), whereby data can be written to several media (hard disks/tapes) simultaneously. In some environments, for example, Oracle, individual tablespaces or files are backed up on separate drives at the same time. Overall performance is better than when you use a single drive.

If you have a sufficient number of tape drives that can be used in parallel, the bottleneck can be shifted from the tape drives to another component. For this reason, you also need to take account of the performance of other subsystems if you want to use the parallel backup option. These subsystems include the controller, CPU, and I/O bus. In many configurations, the controller or bus represents the limiting factor.

Restoring a Parallel Backup [!]

When you restore a parallel backup, you need to be able to read all media in the set. If a single tape is damaged, the backup can't be used. The more tapes you have in a set, the higher the risk of one of them being damaged.

Backup to Hard Disk before Backup to Magnetic Tape

The backup on hard disks and then on tape is the fastest method to back up a database. The backup to hard disk is usually faster than the backup to tape. With this method, you can quickly save several identical copies to hard disks and, for example, store some in external enterprise locations and others at your own site.

As soon as the backup to the hard disk is complete, the impact on system performance is minimal. Because the backup to tape is made from the copy already

made on the hard disk rather than from the production database, there are no competing drains on resources from the backup and database activities. During a disaster recovery process, the data can ideally be restored from the backup on the hard disk. However, this method also has a number of disadvantages:

- You require additional hard disk space equivalent to the size of the database. If your database is large, this may give rise to immense additional costs.
- Until the backup to tape is completed, you have no protection against the risk of potential disasters occurring in the data center. In a disaster recovery scenario, you must recover the files on the hard disk first and then restore the database from the hard disk.

Other options for faster backups are also available, for example, *high availability (HA)* or modern *snapshot procedures*. However, a discussion of these options falls outside the scope of this book.

6.3.3 Restore Performance

The performance requirements for a restore are more important than those for a backup. The restore performance determines when the system will be available again and how quickly business can be resumed. Your objective in this regard is to restore the database and corresponding files quickly and make the system generally available as soon as possible.

The measures to enhance backup performance that we outlined previously also essentially result in shorter restore times. You can therefore examine these proposals from the point of view of both backup and restore performance, for example:

- **Dedicated drives**
  Together with a parallel backup, restoring files and tablespaces to individual, dedicated drives accelerates the process considerably. Only one tablespace or file is written to the drive. As a result, competition for drive resources is avoided.
- **RAID systems**
  RAID 0+1 is faster than RAID5, although these speeds depend on the hardware used. In more cases, the calculation of parity data for the parity drive (RAID5) is more time-consuming than writing the data twice (RAID 0+1). This option is

costly because the usable capacity only amounts to 50% of total capacity, which is significantly less with RAID5:

  - *RAID 0+1 = [single_drive_capacity × (number_of_drives/2)]*
  - *RAID5 = [single_drive_capacity × (number of drives – 1)]*
- **Drives with better write performance**
  You can generally read data more quickly from modern drives that offer a higher write performance. Enhanced reading capacity reduces the time required to perform the restore.
- **Drive array systems with better write performance**
  The benefit of a faster single drive also applies to drive arrays: As a rule, read speeds generally improve in tandem with write performance, which reduces the time required for a restore.

Measures to improve backup performance are often viewed by management as not being particularly urgent. The reason for this is that backups are usually performed out of core business hours and that enhanced performance isn't usually obvious to users. As a result, it can be difficult to obtain the additional means required for modern technology.

However, if you make the argument that clear time savings can be made in terms of the restore process, you may find that your pleas no longer fall on deaf ears. After all, you'll be able to ensure that the system is available after a disaster or emergency much sooner thanks to this technology.

6.4 Summary

The information provided in this chapter was intended to help you develop a backup strategy for your SAP systems, based on your enterprise's business framework. You can protect your systems from the worst-case scenario by combining full and incremental database backups, as well as by backing up your transaction logs and operating system files. You should aim to be able to restore the system completely within a short space of time should such a scenario arise.

The next chapter, Chapter 7, provides additional specific instructions for managing a disaster situation. Chapter 8 introduces you to the SAP-internal database tools, which you can use for automatic backups.

Even the most conscientious system administrator can become overwhelmed when faced with a system failure, a loss of data, or destruction caused by a natural disaster. For such situations, it's always good to have a plan of action and not to be caught completely off guard. This chapter suggests ways in which you can brace yourself for a disaster and prepare for a subsequent system recovery.

7 Disaster Recovery

Thousands of business processes occur on a daily basis and usually without any problems whatsoever. However, even a very brief system outage can seriously disrupt business processes and result in a loss of time, money, and resources. It's therefore advisable to plan for emergency situations so that you aren't entirely helpless when faced with such problems, irrespective of their size and complexity.

This chapter discusses a system administrator's most important task, namely *disaster recovery*, which is a form of system recovery (see Chapter 6).

7.1 Preliminary Considerations

The goal of disaster recovery is to restore the system after an *emergency* in such a way that the enterprise can continue its business processes. Because business processes come to a standstill not only during the system failure itself but also during system recovery, disaster recovery must be performed as quickly as possible. For this reason, it's even more important to have a tried-and-tested recovery plan. Furthermore, the earlier you start to plan, the better prepared you'll be in an actual emergency.

Note on the Following Explanations [!]

This chapter isn't a guide to disaster recovery. Instead, its sole purpose is to increase your awareness of disaster recovery and to stress how important it is to develop a plan.

An emergency is anything that will damage an SAP system or cause a system failure. This includes damage to a database (e.g., accidental loading of test data into a production system), a serious hardware failure, or a complete loss of the SAP system and the infrastructure (e.g., as a result of a natural disaster or fire). In the event of such an emergency, the most important task of the system administrator is to successfully restore the SAP system. Above all else, however, the administrator should ensure that such an emergency doesn't occur in the first place.

A system administrator should be prepared for the worst and have suitable "emergency plans" in place. Disaster recovery isn't the time to try out something new because unwelcome surprises could ruin the entire recovery process.

When developing a plan, ask yourself the following questions:

- If the SAP system fails, will the entire business process fail?
- How high is the loss of earnings, and how high are the resulting costs during a system failure?
- Which important business functions can no longer be performed?
- How are customers supported?
- How long can a system failure last before an enterprise is incapable of conducting business?
- Who will coordinate and manage a disaster recovery?
- What will users do while the SAP system is down?
- How long will the system failure last?
- How long will it take to restore the SAP system?
- Which SAP system components need to be restored so that a remote recovery is possible?

Careful planning will ensure that you're less stressed in the event of an emergency because you'll already know that the system can be restored and the length of time it will potentially take to perform this system recovery.

If you discover that the time required for a system recovery is too long, and the associated losses are too high, management should consider making an additional investment in equipment, facilities, and personnel. Even though a *high availability* (HA) solution is often costly, these costs may not be as high as those associated with possible losses incurred during a disaster.

7.2 Planning for an Emergency

Creating a disaster recovery plan is considered a large project because development, testing, and documentation require a great deal of time, possibly more than a year. The documentation alone may be very extensive, possibly comprising several hundred pages.

Seek advice from experts if you don't know how to plan for an emergency. A plan that doesn't work is worse than having no plan at all because poor planning lulls an enterprise into a false sense of security. Third-party disaster recovery consultants and suppliers can support you during disaster recovery planning.

7.2.1 Which Measures Apply to Disaster Recovery?

The requirements for disaster recovery can be derived directly from the requirements for system availability, which are laid down by management. The guidelines for the requisite system availability are based, for example, on the losses that an enterprise is expected to incur in the event of a system disaster. The monetary loss is usually calculated by management and specified in USD per time unit, while the failure costs depend not only on the enterprise or sector (e.g., industry/public administration) but also on the division in which the software is used (e.g., production/purchasing).

The desired system availability is usually agreed upon in *Service Level Agreements* (SLAs) that you, as an administrator, must fulfill. Therefore, from your perspective, it's also important to know which investments (e.g., for technical equipment or service personnel) are needed to ensure a certain level of availability for the relevant system. Note that the higher the recovery costs, the less time it will take to perform a recovery. However, you can influence these costs through preventive measures (see Chapter 10) and a good recovery plan.

When it comes to technical business units, you must bear in mind that HA comes at a price. If savings are made in the wrong areas, you could be in for a rude awakening. Such costs must be included in the administrative or IT budget.

Financial Effects of a Disaster **[Ex]**

The following discusses three examples of how to calculate financial and entrepreneurial impacts for a disaster:

- **Example 1**
  When forecasting the monetary loss associated with a system failure, your enterprise discovered that transaction data can only be lost for a period of one hour. The resulting costs assume that 1,000 transactions (entered in the SAP system and not restored) will be lost each hour. Such a loss in transactions can lead to a loss in sales as well as extremely annoyed customers. If orders urgently required by customers disappear, the situation can become critical. In this case, you must ensure that the frequency with which data is backed up is sufficiently high (e.g., an hourly backup of the transaction logs).
- **Example 2**
  In your enterprise, you discovered that a system can't be offline for more than three hours. The resulting costs (e.g., at an hourly rate of USD $20,000) are based on the fact that no sales can be posted. In this case, you require a sufficiently efficient emergency strategy or infrastructure to ensure that the system is operational again within three hours.
- **Example 3**
  In the event of an emergency (e.g., the loss of a building that houses the SAP data center), the enterprise can only survive a downtime of two days. After two days, customers start to conduct their business elsewhere. Consequently, an alternative method must be found to continue business (e.g., an alternative data center is built, or an emergency contract is agreed upon with an external provider).

7.2.2 When Should the Disaster Recovery Procedure Begin?

For each disaster recovery plan, you must use a unique set of criteria to determine when such a plan will come into effect and when the procedure will begin. Ask yourself the following questions:

- Which characteristics define an emergency?
- Have these characteristics been fulfilled in the current situation?
- Who must be consulted to assess the situation? The relevant person should know not only how a failure can impact the business process but also be aware of the problems associated with a recovery.

These considerations should help you decide whether or not to initiate your disaster recovery procedure. Alternatively, form a committee that will contribute and assess all of the information required to make a decision within the shortest possible time as well as make a decision in relation to implementing the recovery procedure.

7.2.3 Expected Downtime

Downtime is the period during which a system is unavailable. Even though you can only estimate downtime, it's usually longer than the restore time because, after a system recovery, some tests must be performed, user master records must be unlocked, and notifications must be sent, among other things. It's even more important to have an accurate idea of the restore time.

During downtime, it isn't possible, for example, to process orders or dispatch products. The resulting losses are just one part of the costs associated with a disaster recovery. To minimize disruption, you need to examine alternative processes that can be used while the SAP system is being restored.

During downtime, the following factors generate costs:

- The time during which the SAP system can't be used. The longer the system doesn't work, the longer it will take, after a successful recovery, to make up for the losses incurred during downtime. The transactions from the alternative processes deployed during downtime must be fed into the system to update it. This situation may be problematic in an environment that has extensive transactions.
- A failed system generates more costs than an operational system because additional technology or personnel must be used.
- Customers who can't be served or supported by the enterprise may conduct their business elsewhere.
- If follow-up processes also come to a standstill, your customers may have a claim for recourse.

What is deemed to be an acceptable downtime depends, to a large extent, on the enterprise and the nature of its business.

7.2.4 Restore Time

Restore time is the time required to restore lost data and system operability. Different emergency scenarios have different restore times, depending on the operational needs (e.g., the volume of data to be restored).

The restore time must be adapted to the requirements of the enterprise. If the current restore time exceeds the time limit for these measures, the relevant

managers must be informed of this disparity. Such a disparity can be resolved as follows:

- By investing in equipment, processes, and facilities that will shorten the restore time
- By changing the requirements of the enterprise so that longer restore times are possible

[Ex] **Minimizing Restore Times via Additional Resources**

In an enterprise, it would take a week to restore the system if just one employee was entrusted with this task. The enterprise can't afford the resulting costs or losses in revenue because, during this time, customers would conduct their business elsewhere, vendor invoices would fall due, and invoices would not be paid. In such situations, the management would have to provide additional resources to reduce the restore time to an acceptable level.

If you don't test your recovery procedure (see Section 7.8), the required restore time simply remains an estimate. Use basic testing to ensure that, in the event of an emergency, you can accurately state how much time a system recovery will require (assuming that you have a broad range of experience in this area). You can then also make more accurate statements (to the users) in relation to the expected downtime.

7.2.5 Communication in the Event of a Disaster

A communication concept should form part of your emergency plan. Even if a system failure is usually very noticeable, users can find it annoying if they are left in the dark about their situation.

In certain enterprise areas, a system failure may cause the entire operation to come to a standstill. However, those responsible can't respond appropriately if they aren't informed about when the system is expected to be available again.

If necessary, discuss the following factors with end users:

- Who is affected in the event of a system failure?
- What are the implications of a system failure for the user departments, or which particular dependencies arise?
- What is the timeframe during which information about the system failure must be imparted?

- Which information should be provided (e.g., type, cause, and extent of the disruption, and anticipated downtime)?
- Which contact persons should be informed?
- How should the information be conveyed, or what are the chains of communication?
- Which paths of communication are still available in the event of a disaster? How does communication occur if, for example, the email system is also down?
- After a system has been restored, how do we convey that the system is available again?
- To what extent is information about incident analysis and processing conveyed?

Actively incorporate communication into your recovery plan and coordinate this with the user departments. Good communication can have a calming effect in the event of an emergency because you don't have to deal with complaints and can instead concentrate on restoring the system.

7.3 Recovery Team and Role Distribution

Several people, known collectively as the *recovery team*, are usually involved in a system recovery. A highly coordinated team is the secret to implementing disaster recovery as quickly and as efficiently as possible. There are four key roles within a recovery team:

- **Recovery manager**
  The recovery manager, who coordinates all activities, is responsible for the complete technical recovery.
- **Communications officer**
  The communications officer looks after the users (by telephone, email, etc.) and informs upper management about the current recovery status. If one person assumes responsibility for all communication, the rest of the group can devote themselves to the actual recovery procedure without any interruptions.
- **Technical recovery team**
  This team works to restore the system. If the original plans need to change during the recovery, the technical recovery team must manage such changes and coordinate the technical system recovery.

- **Test and acceptance manager**
  After a recovery has taken place, the test and acceptance manager coordinates and plans the test and acceptance procedures.

The number of employees who assume these roles varies depending on the size of the enterprise. In a small enterprise, for example, one person can assume the role of recovery manager and communications officer. In addition, the descriptions and range of tasks will most likely vary depending on the needs of your enterprise.

Structure your disaster recovery concept in such a way that each team member and each role knows exactly which tasks are to be performed and when. Describe the dependencies and coordination processes between the roles, and create checklists for each team member.

[+] **Status Notice**

To prevent incidents involving employees working on the recovery, we recommend that you create a status notice. Key points in the recovery plan are listed here as well as estimates in relation to when the system will be restored and operational again.

Also bear in mind that key employees may not be available in the event of an emergency (e.g., due to vacations or sick leave). Therefore, the team must also be able to perform a successful recovery without these people. In an actual emergency, this issue can be very urgent.

[!] **Planning with Employees from Other Locations**

If the emergency is a major natural disaster, your on-site employees will be extremely concerned about their own families as well as the enterprise itself. In some cases, key employees may be badly or even fatally injured. You should also prepare for such situations and formulate plans accordingly. Allow for the fact that employees would have to be flown in from other locations and integrated into the recovery team.

7.4 Types of Disaster Recovery

Disaster recovery scenarios can be divided into two types:

- **In-house recovery**
  In-house recovery is disaster recovery that you perform yourself at your enterprise location. The in-house infrastructure must remain intact as far as possible

(this is usually the case). Ideally, the recovery is made using the original hardware. In the worst-case scenario, the original hardware must be replaced with a backup system.

- **Remote recovery**
  Remote recovery is disaster recovery performed at a special disaster recovery location. In this scenario, the entire hardware and infrastructure has been destroyed as a result of a fire, flood, earthquake, or similar. Consequently, the new servers have to be configured from scratch.

In the case of a remote recovery, you must bear in mind that a second system recovery must take place at the original location as soon as the original facility has been rebuilt. Plan and schedule the second recovery in such a way that as few users as possible are inconvenienced by the fact that the system won't be operational during this recovery.

7.5 Emergency Scenarios

Although numerous emergency scenarios are conceivable, it's impossible to develop plans for all possible scenarios. Therefore, to keep this task manageable, you should limit yourself to approximately three to five probable scenarios. If an emergency occurs, you can adhere to the scenario that best corresponds to the actual emergency. An emergency scenario comprises the following points:

- Description of the emergency
- Planning the main tasks at a high level
- Estimated downtime

The best way to prepare for an emergency is to use emergency scenarios:

1. Use Section 7.5.1 through Section 7.5.3 as a starting point, and prepare three to five scenarios that cover the largest possible range of emergencies.
2. For each scenario, create a plan for the main tasks at a high level.
3. Test the planned scenarios by simulating different emergencies and checking whether your scenarios could be applied to the actual emergency.
4. If this isn't the case, change the scenarios or develop new ones.
5. Repeat the process.

The following three examples are arranged in order of increasing severity. Note that the downtimes cited are merely examples to illustrate the situations you may encounter. Your own downtime will differ from those specified here. You must therefore replace the sample downtime with a downtime that applies to your environment. It will become clear that, depending on the specific emergency, various extensive measures must be taken and that extremely long downtimes may occur even if the damage appears to be minor.

7.5.1 Damaged Database

A database may be damaged if test data are inadvertently loaded into the production system or if data incorrectly transported into the production system causes a crash. If such an incident occurs, the SAP database and associated operating system files must be restored. The downtime is, for example, four hours.

7.5.2 Hardware Failure

The following hardware can fail:

- Processors
- Hard disks or their control unit
- RAID controller (known as an array failure)

If such a failure occurs, the following steps are necessary:

1. Replace the failed hardware.
2. If required, rebuild the server (operating system and programs).
3. Restore the SAP database and associated files.

The downtime is, for example, three days, broken down as follows:

- Two days to procure replacement hardware
- One day to rebuild the server (by one person), that is, eight working hours in total

[+] **Planning a Production Server Replacement**

Plan and test the use of your test system (QAS) as a backup server if the production server (PRD) fails.

7.5.3 Complete Loss or Destruction of the Server Facility

The following components may be destroyed if a catastrophe occurs:

- The servers
- The entire supporting infrastructure
- All of the documentation and materials in the building
- The building itself

Such a complete loss of facilities may be the result of a natural disaster such as a fire, flood, hurricane, or a manmade catastrophe. If such a catastrophe occurs, the following steps are necessary:

1. Replace the destroyed facilities.
2. Replace the destroyed infrastructure.
3. Replaced the destroyed hardware.
4. Rebuild the server and the SAP environment (hardware, operating system, database, etc.).
5. Restore the SAP database and associated files.

The downtime is, for example, eight days, broken down as follows:

- At least five days to procure the hardware. If it's a regional catastrophe, it may take longer to procure the hardware because vendors may also be affected by the catastrophe.

National Vendors [+]

Turn to national vendors that have several regional distribution centers. As an additional backup measure, you should look for alternative vendors in distant regions.

- Two days to rebuild the server (by one person), that is, 16 working hours in total.
- While the hardware is being procured and the server is being rebuilt, an alternative facility in which a minimal emergency network can be constructed must work. The integration into the emergency network may take one day, for example.

A complete loss makes it necessary to perform a recovery in a new facility or in a different building. Depending on the size of the enterprise, how important the

SAP system is for the enterprise's business processes, and the regional risk of a natural disaster, it may make sense to build a redundant data center. If one of your data centers is destroyed, operation of the system landscape can then switch to the other data center. However, both of these data centers must be built at least a few kilometers apart from each other. If housed in the same building, it's highly likely that both data centers would fail in the event of a disaster.

If your enterprise doesn't have or want to use the resources necessary for a redundant data center, you can agree on a contract for a disaster recovery location with an external provider. Then, if a disaster occurs, the provider's hardware will be available for your emergency use.

[!] **Recovery Location in an Emergency**

Having a contract for a disaster recovery location doesn't guarantee that this location will be available in the event of an emergency. If a catastrophe that affects an entire region occurs, many other enterprises will want to access the same disaster recovery locations as you. In such a situation, you may have to cope without a recovery location because other enterprises will have booked the location before you.

Sometimes, the equipment in a disaster recovery location or emergency data center isn't as efficient as your production system. Therefore, when making plans, bear in mind that you'll be faced with lower performance and limited transactions. For example, reduce background jobs to only the most urgent jobs. Alternatively, only grant recovery system access to those users who need to perform essential business tasks.

7.6 Recovery Script

A recovery script is a document that contains step-by-step instructions for the following aspects:

- The procedure for restoring the SAP system
- The individuals responsible for each step
- The estimated time required for lengthy steps
- The interdependencies between steps

A script helps you implement suitable steps for restoring the SAP system and avoids the risk of any steps being omitted. If you inadvertently omit an important

step, you may have to start the entire procedure from scratch, thus delaying the system recovery.

To create a recovery script, you need the following:

- A checklist for each step
- A document that contains screenshots that explain the instructions (if required)
- Flow charts if the sequence in which the steps or activities are performed is complex or confusing

If the main person responsible for the recovery is unavailable, a recovery script will help his representative fulfill this task. The script must therefore fully describe all tasks in an easy-to-understand manner.

Important Steps in the Recovery Procedure

If you want to shorten the recovery process, you can define a procedure whereby as many tasks as possible are handled concurrently. Provide a schedule for each step. The most important steps are as follows:

1. During an emergency, you can support the recovery by doing the following:
   - Gather facts.
   - Retrieve the backup tapes from the remote storage location.
   - Have the crash kit ready (see Section 7.7).
   - Notify all relevant employees (e.g., the in-house SAP team, key users affected by the emergency, infrastructure support, IT, facilities, on-call consultants, etc.).
   - Prepare functional organizations (sales, accounting, and shipping) for alternative procedures for important business transactions and procedures.
   - Notify non-SAP systems that have interfaces from and to the SAP system about the system failure.
2. Minimize the effects of the failure by implementing the following measures:
   - Stop all additional transactions into the system (e.g., interfaces from other systems).
   - Collect transaction documents that will have to be entered again manually.

3. Start the planning process by implementing the following measures:
   - Analyze the problem.
   - Select the scenario plans that best correspond to the emergency that has actually occurred.
   - Change the plans, if necessary.
4. Decide when the disaster recovery procedure will begin:
   - Which criteria formed the basis for determining that an emergency had occurred? Were these criteria satisfied?
   - Who makes the final decision in relation to confirming that an emergency has occurred?
5. Ascertain whether an emergency has occurred.
6. Implement the recovery procedure.
7. Test and approve the restored system. Key users should conduct the relevant tests. Such users use a checklist to clarify whether the system has been restored to a satisfactory level.
8. Update the system with transactions that alternative processes handled during the system failure. As soon as this step is complete, the outcome should be approved again.
9. Notify users that the system is operational again.
10. Arrange a postmortem meeting to ascertain why the disaster occurred.
11. Assess the recovery team's experience of the system recovery, and optimize your disaster recovery plans accordingly.

The recovery script must be easily accessible in the event of a disaster. It must not be stored on a server that may no longer be accessible in the event of a network failure. Also bear in mind that a paper copy could be destroyed in a fire. Prepare yourself for such emergency scenarios and store the recovery script redundantly. Make sure that the storage location is widely known and accessible to those individuals responsible for a recovery in the event of an emergency.

Dependency on Other Applications

Your SAP system is usually connected, via interfaces, to other upstream or downstream systems. If the SAP system fails, feeder systems may also come to a standstill because RFCs accumulate en masse and can't be processed. In addition,

downstream systems may not work because your SAP system doesn't make the necessary data available. You can therefore see how easy it is to experience a chain reaction that will have far-reaching consequences for the system landscape and business processes.

In your recovery script, give some thought to communication with those individuals responsible for the applications connected to the SAP system. Make sure that the interfaces are stopped or stop them yourself. Decide how the data will be resynchronized after a system recovery.

7.7 Crash Kit

A *crash kit* contains everything you need to rebuild the SAP server, reinstall the SAP system, and restore the SAP database, including all related files. You must therefore store everything you need to restore your SAP environment in one or more containers physically—in the form of backup tapes, hardware, and documents—and/or digitally. If your location needs to be evacuated, you won't have any time to gather everything you need at the last minute.

You should therefore check your crash kit regularly and check whether all of the elements are still up to date and operational. A service agreement is a good example of a crash kit component that requires such a regular check. If the agreement is no longer valid because its validity period wasn't extended in time, you may not be able to access the services provided by external providers in an emergency, or you may have to enter into negotiations first.

Updating the Crash Kit [!]

If a (hardware or software) component on the server is changed, replace the obsolete component in your crash kit with the latest, tested element.

The crash kit should be stored in a room separate from the servers. If the crash kit is stored in the server room, the crash kit will also be affected if servers are lost. Examples of suitable storage options include the following:

- A commercial data storage location outside the enterprise's location
- Other enterprise locations
- Another secure part of the building

Next, we'll name the most important items that should form part of any crash kit. You can add or omit items, depending on your particular environment. The inventory is sorted according to documentation and software.

Documentation

A crash kit must contain the following documentation:

- A disaster recovery script.
- A test and verification script for functional user groups, which is used to ascertain the functionality of the restored system.
- Installation instructions:
  - Operating system
  - Databases
  - SAP system
- Special installation instructions for the following:
  - Drivers that must be installed manually
  - Programs that must be installed in a certain way
- Copies of the following:
  - SAP licenses for all instances
  - Service agreements (with telephone numbers) for all servers

[!] **Checking the Validity of the Service Agreements**

Make sure that the service agreements are still valid. You should perform this check regularly.

- Instructions for retrieving backup tapes from external data stores outside the enterprise's location.
- A list of individuals authorized to retrieve backup tapes from data stores outside the enterprise's location. This list must correspond to the list available at the external data store.
- A parts list that contains enough information to ensure that new hardware can be purchased or leased if the server is destroyed. After a certain length of time, original parts may no longer be available. You should then draft an alternative parts list. At this time, you should also give some thought to updating your equipment.

- Layout of the file system.
- Layout of hardware.
- Telephone numbers of the following:
  - Key users
  - Information service employees
  - Facilities personnel
  - Other infrastructure personnel
  - Consultants (SAP, network, etc.)
  - SAP hotline
  - Data stores outside the enterprise's location
  - Security department or security employees
  - Contact partners within the framework of service agreements
  - Hardware vendors

Software

The crash kit should contain all of the software components required to completely rebuild a server.

- Operating system:
  - Installation kit
  - Hardware drivers not contained in the installation kit (e.g., network cards or SCSI controllers)
  - Service packs, updates, and patches
- Database:
  - Installation kit
  - Service packs, updates, and patches
  - Recovery script for automating a database recovery
- SAP system:
  - New installation files of the SAP release used and the database
  - Currently installed kernel
  - System profile files

- *tpparam* file
- *saprouttab* file
- *saplogon.ini* files (for SAP GUI)

- Other programs integrated into the SAP system (e.g., a control package)
- Other software for the SAP installation:
  - Auxiliary programs
  - Backup
  - UPS control program
  - Hardware monitor
  - FTP client
  - Remote control program
  - System monitors

[+] Crash Kit Inventory

The person who seals the crash kit should also compile a signed and dated inventory. If the seal is broken, you must assume that some items have been removed or changed, and, as a result, the kit could be completely useless in an emergency.

7.8 Testing the Disaster Recovery Procedure

By simulating a disaster recovery, you can ensure that your system can actually be restored and that all of the tasks listed in the disaster recovery plan can be executed. By performing a simulation, you can ascertain whether the following are true:

- Your disaster recovery procedure works
- Changes have occurred, steps haven't been documented, or the necessary updates haven't been performed
- Some steps require additional explanation
- Steps that are quite clear to the person writing the documentation are also clear to other individuals
- Older hardware is no longer available

If one of these scenarios arises, you must revise your recovery plan. You may also have to upgrade your hardware so that it's compatible with the equipment currently available. Furthermore, you should draft an alternative procedure in response to inconsistencies that previously went unnoticed in an emergency.

Because many factors influence the actual recovery time, it can only be determined through testing. As soon as you have actual time values instead of estimates, your emergency plan will become credible. If the procedure is practiced on a regular basis, everyone will know what to do in an emergency, thus making it possible to avoid the worst-case scenario.

To test your disaster recovery procedure, follow these steps:

1. Implement your disaster recovery plan in a backup system or at a remote location.
2. Envisage a random emergency scenario.
3. Implement your emergency plan to see if it's effective in such a situation.
4. Perform disaster recovery at the same location where it will occur in the event of an emergency. If you have more than one recovery location, run the tests in each of these locations. The equipment, facilities, and configurations may differ from location to location. Document all of the steps that need to be executed at each location. You're now immune to not being able to restore the system at a certain location in the event of an emergency. Other options for locations where you can test your disaster recovery scenario include the following:
   - A backup server at your location
   - Another enterprise location
   - Another enterprise with which you have a mutual support agreement
   - An enterprise that provides disaster recovery locations and services

During a real disaster recovery, your permanent employees will carry out the relevant tasks. However, you should take precautions in case some of your key employees are unavailable during the disaster recovery. A test procedure can therefore include the random selection of an individual who won't be available and won't participate in the test procedure. This procedure reflects a real situation in which a key employee is absent or has been seriously injured, for example.

Furthermore, employees from other locations should also participate in testing. Integrate these individuals into the tests because you may also require them

during a real disaster recovery. These employees can fill the gap arising from unavailable personnel.

At least once a year, you should run through your disaster recovery from start to finish. However, the frequency with which you do this is a commercial decision that should be made while considering the costs involved.

[!]

Maintaining the Production System

Note that during disaster recovery testing, employees are still needed to maintain the real production system.

7.9 Minimizing the Risk of Failure

There are many ways to minimize the risk of failure. Some of the suggestions listed here may seem obvious. In reality, however, they are frequently ignored.

7.9.1 Minimizing the Risk of Human Error

Many emergencies are triggered by human error (caused, for example, by an exhausted operator). For potentially dangerous tasks (such as deleting the test database, moving a file, or formatting a new drive), a script should be created with a checklist that can be used to verify the individual steps.

[+]

Critically Assessing Your Own Capabilities

Don't perform any dangerous tasks if you feel tired. If you nevertheless have to do it, seek a second opinion before you start.

7.9.2 Minimizing Single Points of Failure

A *single point of failure* occurs when the failure of a single component causes the entire system to fail. You can minimize the risk as follows:

- Ascertain the situations in which a single point of failure can occur.
- Devise a forecast of what happens when this component or process fails.
- Eliminate as many single points of failure as possible.

Single points of failure may include the following:

- The backup SAP server is in the same data center as the production SAP server. If the data center is destroyed, the backup server will also be destroyed.
- All SAP servers are connected to the same power supply. If the power supply is interrupted, this affects all of the equipment connected to this power supply. In other words, all servers crash.

Cascading Failures

A *cascading failure* occurs if one failure triggers a whole series of failures, thus making the problem even more complex. In this case, the recovery comprises a coordinated solution for numerous problems.

Cascading Failure [Ex]

The following is an example of a cascading failure:

- A power outage that affects the air conditioning unit can cause the air-conditioning controls in a server room to fail.
- If the server room can't be cooled, the temperature in the room rises above the permissible operating temperature for the equipment.
- Overheating causes a hardware failure on the server.
- The hardware failure causes damage to the database.
- Overheating can also affect numerous other pieces of equipment and systems (e.g., network devices, the telephone system, and other servers).

A system recovery after a cascading failure can be complex because, when solving one problem, you may discover other problems or other damaged pieces of equipment. Alternatively, some equipment can't be tested or repaired until other pieces of equipment become operational again. In the air-conditioning example, a system could monitor the air-conditioning unit or the temperature of the server room and notify the relevant employees when a certain threshold value is exceeded.

7.10 Continuing Business During a System Recovery

During disaster recovery, any affected business processes must continue as soon as possible to avoid or minimize an enterprise's financial losses. Give some thought to which alternative procedures can support key business processes when an SAP system fails, for example:

- Collection of cash
- Order processing
- Product shipping
- Invoice payments
- Payroll
- Alternative locations for continuing business

If there's no alternative process, your business operations will decline or come to a complete standstill, which may result in the following problems:

- Orders can't be entered
- Products can't be shipped
- Cash can't be collected

The following alternative processes are conceivable:

- Manual data entry in paper form (e.g., handwritten purchase orders)
- Working on standalone PC systems

Together with your end users, plan how certain business processes can continue to run during a system recovery. Define when or during which expected downtime an alternative process will enter into force. Furthermore, give some thought to how data generated during the emergency process can be transferred to the SAP system after the system recovery.

7.11 Summary

Disaster recovery is a special type of system recovery that requires proper advance preparation. An in-depth concept, the necessary tools, and planned testing on a regular basis will all contribute to helping you prepare for an emergency. This chapter helps you think of everything you need.

Calculate the costs that your enterprise and systems would incur as a result of a system failure or the losses that would arise if a system were unavailable for a period of one hour. Concrete figures are the easiest way to convince everyone of the need to invest in disaster recovery.

Contents

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

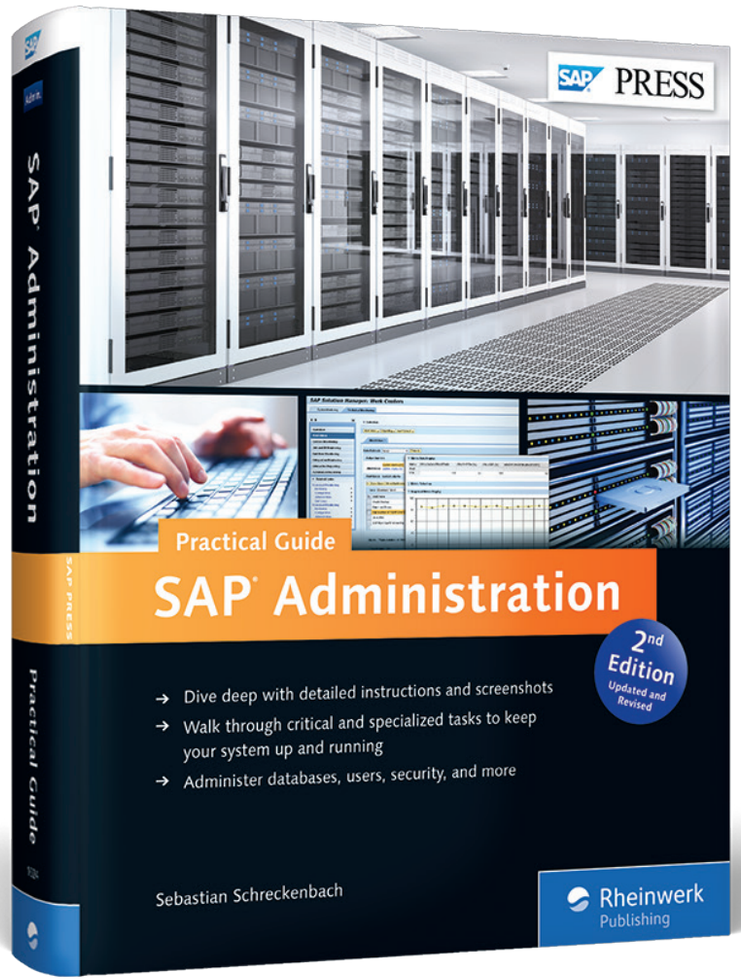
T

U

V

W

X



Sebastian Schreckenbach works as a senior consultant in SAP Basis administration at Steria Mummert Consulting AG in Leipzig, Germany. Prior to this, he worked as an SAP Basis administrator for many years in the SAP Competence Center in Dresden, Germany.

Sebastian Schreckenbach

SAP Administration—Practical Guide

912 Pages, 2015, $79.95/€79.95
ISBN 978-1-4932-1024-4

www.sap-press.com/3639