

SAP - Audit Guidelines R/2

Release 5.0E
As of 04/01/1995
Material-no. 5001 9057
Fax-no. 0049-6227-74-1497



SAP Audit Guidelines R/2

	Introduction.....	
1	System overview.....	
1.1	Objective.....	
1.2	Requirements.....	
1.3	Risks.....	
1.4	Auditing steps.....	
1.5	Proposed authorizations for the auditor.....	
2	Security and access protection.....	
2.1	Objective.....	
2.2	Requirements.....	
2.3	SAP facts.....	
2.4	Risks.....	
2.5	Auditing steps.....	
3	System maintenance and release procedures for source programs and job runs.....	
3.1	Objective.....	
3.2	Requirements.....	
3.3	SAP facts.....	
3.4	Risks.....	
3.5	Auditing steps.....	
4	A B A P.....	
4.1	Objective.....	
4.2	Requirements.....	
4.3	SAP facts.....	
4.4	Risks.....	
4.5	Auditing steps.....	
5	Computer center job submission procedure, job documentation and computer center logs.....	
5.1	Objective.....	
5.2	Requirements.....	
5.3	SAP facts.....	
5.4	Risks.....	
5.5	Auditing steps.....	
5.6	SAP job documentation (sample proposal).....	
6	Procedure for changing tables.....	
6.1	Objective.....	
6.2	Requirements.....	
6.3	SAP facts.....	
6.4	Risks.....	
6.5	Auditing steps.....	

7	Batch input interfaces.....	
7.1	Objective.....	
7.2	Requirements.....	
7.3	SAP facts.....	
7.4	Risks.....	
7.5	Auditing steps.....	
8	Master data changes.....	
8.1	Separation of functions.....	
8.2	Traceability.....	
9	Posting data reconciliation.....	
9.1	Objective.....	
9.2	Requirements.....	
9.3	SAP facts.....	
9.4	Risks.....	
9.5	Auditing steps.....	
10	Day-end/Month-end closing.....	
10.1	Objective.....	
10.2	Requirements.....	
10.3	SAP facts.....	
10.4	Risks.....	
10.5	Auditing steps.....	
11	Year-end closing.....	
11.1	Objective.....	
11.2	Requirements.....	
11.3	SAP facts.....	
11.4	Risks.....	
11.5	Auditing steps.....	

Introduction

This R/2 auditing guide is intended to provide useful tips for auditing SAP systems to external auditors, information systems analysts, and employees of internal audit departments. The guide is intended as a 'recommendation', and not as a 'binding guideline' or 'norm'. Any and all responsibility for the type, scope and results of external and internal audits lies with the auditor himself.

The guide will also be available for the R/3 system environment, Release 2.2D, in June 1996. Both guides will also appear in Lotus Notes for SAP and Business Partners under 'Main Topics' -> Internationalization.

In order to properly study this manual, you should have a good knowledge of the SAP system, and you should also be familiar with your own country's standards for auditing computer-based accounting systems.

The authors are members of a team from the SAP Auditing study group. Their experiences are presented here for your benefit.

Copyright 1995:

Mr. Balser	Hermes Kreditversicherungs-AG, Hamburg
Mr. Barthel	FORBIT e.V., Hamburg
Mr. Bernd-Striebeck	KPMG, Duesseldorf
Mr. Goettmann	Philipp Morris GmbH, Muenchen
Mr. Grotebrune	Unilever International Audit, Hamburg
Mr. Hoerner	ABB Management Services GmbH, Mannheim
Mr. Hungerbuehler	ATAG Ernst & Young, St.Gallen
Mr. Jackisch	Price Waterhouse, Duesseldorf
Dr. Koch	Osnabrueck/Melle
Mr. Lencses	SCHITAG Ernst & Young Deutsche Allgemeine Treuhand AG, Stuttgart
Dr. Peemoeller	ORIGIN Information Technology GmbH, Hamburg
Mr. Sengpiel	Beiersdorf AG, Hamburg
Mr. Weinkauff	Metallgesellschaft AG, Frankfurt

Each author is exclusively responsible for the contents of the chapters they have written. The manual was edited by SAP AG, Walldorf, financial development department.

Note: This work, and all of its parts, are protected by copyright. Any unauthorized use of this work outside the limits of the copyright is improper and punishable by law. This applies particularly to duplicating, translating into other languages, microfilming, and the storing and processing of the work in electronic systems.

You can find further details on topics covered here in the corresponding SAP manuals for Release 5.0 and 5.0E.

1 System overview

1.1 Objective

1.2 Requirements

1.3 Risks

1.4 Auditing steps

1.4.1 Responsibilities

1.4.2 Systems in production (Test,...,Production)

1.4.3 Setting up an authorization for the auditor

1.4.4 Assessing the business structure

1.4.5 Release versions

1.4.6 Components

1.4.7 Modifications

1.4.8 Autoskip function

1.4.9 Data flow plan

1.5 Proposed authorizations for the auditor

1 System overview

This first chapter of the SAP audit guide provides you with a quick overview of the SAP system and its technical and organizational integration. The auditor needs this overview in order to obtain an adequate system orientation, to be able to assess the overall state of the system, and to determine which audit steps will be required.

1.1 Objective:

The aims outlined above may be broken down as follows:

1.1.1 Obtain a technical and organizational overview of the System

Because of the enormous technical scope of the system and its constant changes, the only people who are capable of obtaining a full overview of the R/2 System are those who concern themselves exclusively with its software. As a rule auditors do not come into this category; often coming from outside, they often have to establish with a few initial steps on the system what technical functionality is employed by a particular user. It should be possible to obtain this technical overview in the shortest possible time and without the need for complex additional technical efforts. Ideally the system should be able to automatically provide all the most important information - "at the press of a button", so to speak - about itself, the system status and (if required) all changes to the latter since a specified time. From an auditing point of view, the main aspects of the system status include:

- source and load programs,
- transactions,
- ABAP programs,
- tables,
- files,
- authorizations, profiles, and user master records
- data carriers, and
- other security measures (such as table categories, separation of different clients).

The R/2 system's organizational integration and the changes made to it determine the effectiveness of technical measures aimed at ensuring efficient data processing. Comprehensive documentation is required to fully explain the system status. The overview is supplemented with sample checking by the users (for example, to examine the user authorizations transaction), the system documentation (for example, to check the program and table documentation), and the system environment (for instance, working with the system in the event of an abnormal termination).

1.1.2 Clarity of the system for the auditor and his/her task

The SAP system is designed for ease of use. This is critical for the auditor, because a system which is easily navigable is a prerequisite for performing an accurate examination, which complies with professional auditing standards, of the system.

1.1.3 Defining the audit scope

Finally, the system overview should enable the auditor to concentrate on specifically defined auditing areas. After gaining an overview, the scope of the audit should be clear to all concerned. You should also be able to predict the functional and chronological framework of the audit at this point.

1.2 Requirements:

The system installation under scrutiny must meet auditability requirements. You can assume that auditability requirements demanded by tax authorities are generally met. In particular, the application of the system and all modifications to the system have to be in keeping with SAP's recommendations (for example, the recommendations given in the standards manual S24.2).

This specifically affects

- the procedures for making changes and adjustments to the standard software;
- documentation of changes to the system and its environment;
- naming conventions when changing transactions, ABAPs, tables and files.

Client number 000 (or any other reserved delivery client) must only be used for the software delivered by SAP, since it serves as a reference for all the other clients from 01 to 99 (or for each of the other clients if a non-delivery client is used), and will be partially overwritten on the next change of release or maintenance level. Certain key information supplied by SAP resides in client 000, and may only be maintained by the system administrator (in consultation with SAP specialists, when necessary).

1.3 Risks:

- inefficient processing
- inconsistent data
- faulty operation
- lack of control
- unreliability.

1.4 Auditing steps:

1.4.1 Responsibilities

You must acquire an overview of the

- critical data and tables,
- authorizations, and
- interfaces,

which should then be expanded in the course of auditing.

1.4.2 Systems in use (testing,...,productive systems)

You need to establish which systems are being deployed, and which of these are being used for productive, development, testing, acceptance or training purposes. If the information provided is unsatisfactory, a check should be made, if necessary outside the SAP System, to see whether other installations are being deployed by the user. This may be the case in the event of release changes or for the parallel testing of different versions (such as is currently the case with various users under 4.3 and Release 5.0). Clarity concerning the different installed systems can be provided, for example, by the automatic log (but see also below).

In the productive system - the focus of this manual - you need to look at **Table 000 in client 000** to see which clients are active in this installation. The first thing to establish is which system in which client contains the

- productive client(s)
- test client(s).

This is illustrated in the following example:

Table display		000 Clients				
Client	Description	F-Cal.	Date	PR	Q	GL-client GL-req.ledger
000	Delivery Cl.	01	T	X	N	SV
001	AFOS-PROD	01	T		N	SV
002	AFOS-TEST	01	T		N	SV

Fig. 1.4-1: Table T000

1.4.3 Creating an authorization for the auditor

The auditor should have direct access to the system, if possible including all authorizations listed in chapter 1.5. In granting access to personal data, care should be taken to ensure that data security requirements and any existing operating agreements are duly complied with.

The setting 'OK Code = S' will ensure that the auditor may not and cannot make any changes to the data.

1.4.4 Assessing the business structure

As a first step, the business structure described by SAP should be noted and analyzed within each productive client (see Table 000 - Fig.1.4-1). SAP provides you with a system that hierarchically classifies the data and access structures into

- client
- company code
- business area
- plant, and
- storage location

to represent these business structures.

The company codes of a given client are listed in Table 001:

Table display		CLIENT 00		Date 08/20/95			
001	Company codes						
CC	NAME				CITY		
	CG	Ctry	Currency	Lang	FV	Alt.accounting period	==> PF2
01	Ball Bearings, Inc.				21210 Baltimore		
	AA	US	USD	E	12		
02	Bicycles, Inc.				19113 Philadelphia		
	AA	US	USD	E	12		
03	FIMA Italia				20041 Agrate Brianza (MI)		
	AA	I	LIT	I	09		

Fig. 1.4-2: Table 001 'Company codes'

Additional cross-linking data are filed in Tables **001G** and **001W**:

Table display			CLIENT 03	DATE 03/18/95
001G Business areas				
CC	Bus.area	Bus.area name		
01	00	Not allocated		
01	01	Construction machinery		
01	02	Frankfurt branch		

Fig. 1.4-3: Table 001G 'Business areas'

Table display													001W	Co.Cde/Plant/Stor.loc.key												
CC	Pt	St	Pst	Code	Customer no.	Plnt	Hrs	R	FC	B	CP	P	I	C	B	B	M	Plnt								
			City		RC CCd CtC		Textno																			
01	01	XX	19113		00000000	WW	24	N	01	Y	00	K	Z	J	_	Z	_	**								
			Raw matls storage	_	_	-	-																			
01	AB	01	21210		00000000	WW	24	N	01	Y	00	K	Z	J	_	Z	_	**								
			Central storage	_	_	-	-																			
01	XX	A1	02215		00000000	WW	-	N	01	_	00	_	_	_	_	_	_	_								
			Office materials	_	_	-	-																			

Fig. 1.4-4: Table 001W 'Company/plant/storage loc. codes'

Consequently, the major SAP-developed databases (such as LIFA for vendors, KUNA for customers, MARA for materials and SKSA for general ledger accounts) have a hierarchical structure. In accordance with the business structure detailed above, these databases are subdivided into segments as follows:

- the **A segment** (client/group level, normally MARA, KUNA, LIFA) contains master data for the client and the client's vendors and customers, i.e. addresses, descriptions, key information etc.
- the **B segment** (company number/company code level, normally MARB, KUNB, LIFB) contains such data as payment parameters, shipping terms etc.
- the **C segment** (transaction data, normally MARC, KUNC, LIFC) contains such details as customer and vendor totals per account, and the accounting period etc.
- the **D segment** (storage location level, normally MARD) contains specific information on the material held at each storage location.

This information forms the basis for checking user authorizations and comparing their specific competences and responsibilities, as well as the technical functionality available to them.

1.4.5 Release versions

This audit manual is based on release version 5.0E. You can determine the particular release version of the application being audited by calling up transaction **TM20**:

Cust.spec. Transport data	Time: 16:23:39	Date: 02/13/95

Transport information		
Customer No.:	0	Distrib. class: 3
Cust. info No.:	00000000	PC connections: 99999
SAP Release	50E	Cust.CMOD active:
Languages	D,E,F,I,N,S,K	
Transport tape iss.dte..:	05/20/94	Time: 15:58:21
OK _ PF: 4=End 7=maintain SAPCMOD 15=Display components 1 - 000		

Fig. 1.4-5: Transaction TM20

In addition to identifying the active Release version, by pressing PF15 within Transaction TM20 (see above), you can call up the active price list components. You can call up the descriptions of the activated price list components for each user system via the function key PF13 within Transaction **TM20**, or view them directly in table **TCOMP**.

1.4.6 Components

The active components of the system mentioned under 1.4.5 above refer to particular functionalities of the SAP system, namely

- | | |
|----------------------------|--|
| - Source and load programs | Modification through Ex-SAP and DR authorization |
| - ABAP/4 programs | TM38, SAPLIMU and Ex-SAP |
| - Jobs | Ex-SAP |
| - Screens | TM51ff, Ex-SAP, SAPDYNU and |
| Screenpainter | |
| - Tables | TM31ff, SAPTABU, ABAPs |
| - Data Dictionary objects | TM61ff, SAPLIMU |

The functionalities listed on the left can be extended and modified by means of the functions listed on the right.

The connection between a user's activated price list components and the functionality supplied by SAP can be called up using the table display for Table **TDIR** and with the aid of the ABAP program **RSDIR003** 'Evaluation of Table DIR'.

1.4.7 Modifications

Modifications by the user should only be made - in line with SAP recommendations - in conformity with the naming conventions. The auditor should first check out these by displaying and printing out all

- transactions that begin with Z or Y from Table STC,
- ABAPs that begin with Y or Z,
- tables that begin with T9.

In a second step, the transactions, reports and tables in the system and in the documentary files or online helps should be examined to check their functions and proper documentation.

The auditor should take samples to test further functions which, from their external appearance, do not seem to have been altered. This can be done by systematically investigating which user made which last change, and on which date. For details, see the list on the following page:

Audit Guidelines	Release 5.0E	Date 04/01/95	Author AK REV	Page I-2
------------------	-----------------	------------------	------------------	-------------

**SAP System
documentation**

System status

Changes

Tables	TM08,OM08,TDIR,DD31, TM63	docu status=USR otherwise:date & last changer in table header
Table fields	DD35	Doc. status= USR in selection screen
ABAP	ABIS, TDIR, RDIR	in ABIS PF13 Report select PF2 date last changer and version number plus addit.info in report doc. or in RDIR
Program fields	DD39	Doc.status=USR in selectn scrn
Transactions	STC, TDIR	generally: date & changer of last STC change in header of STC, STCT
Dynpros	TM51,52, TDIR	Dte/time of Dynpro save/generation & changer in TM51
Dynpro fields	DD40	Doc. status=USR in selectn scrn
Authorizations	TMU4: User overview Profile overvw Authorz.overvw Objects overview	TMU9 changes: -for users -for profiles -for authorizatns
Price list- components	TM20,TDIR,RSDIR003 Transctns/price list- compnts,ABAPs/price header of the STC, STCT pr.lst compnts	general:date & changr of the last STC chng in compnts,screens/
Company org. structure	Client T000 Company codes T001 Business areas T001G Plants T001P, T001W Integration level T001C	date&last changer in table header (PF15 in TM08)
Files	DDIN database tab. DD33	Doc.status=USR in sel. screen
Database fields	PF17 after DD33	Doc.status=USR in sel. screen

1.4.8 Autoskip function

Precautions must be taken to ensure that SKIP logs, resulting from the use of the Autoskip function, are passed to the accounting section. When a SKIP occurs, the originally posted document is missing in the ABEZ file. This means that the accounting clerk must repost the technically "lost" document. The SKIP log (by means of the SAP logband utility) has the attributes of a document in case of such a repeat posting, and as such must be stored for a period of at least six years.

1.4.9 Data flow plan

Incoming data from non-SAP systems needs to be recorded; specifically, its format (as a data record description), content, origin and quantity should be registered.

Data flowing out of the SAP System must also be noted and recorded; the details required in this case are its format (that is, as a data record description and also regarding the file format), content, target system and quantity.

The details should be scrutinized by doing sample checks on authorizations granted to process batch input sessions, as well as download capabilities, if these exist.

1.5 Proposed auditor authorizations

R:ALL with OK code 'S'

This is the highest recommendable level of access; you should also bear in mind all legal regulations regarding data security, as well as any relevant business/contractual agreements which may affect decisions on granting system access.

2 Security and access protection

2.1 Objective

2.2 Requirements

2.3 SAP-Facts

2.3.1 The authorization concept - basics

2.3.2 Collective authorizations and profiles

2.3.3 Separating maintenance and activation

2.3.4 User master

2.3.5 Password protection and logon

2.3.6 Authority Check

2.3.7 Upstream security systems

2.3.8 Table TSTC 'SAP transaction codes'

2.4 Risks

2.5 Auditing steps

2.5.1 User management

2.5.2 Security and access protection

2.5.3 Important individual authorizations

2 Security and access protection

2.1 Objective

An access protection system and the ability to grant individual authorizations basically serves four purposes:

- To protect confidential data against unauthorized disclosure
- To protect the data against unauthorized, but also against unintentional, changes or deletion
- To facilitate the transparency of the procedures by tracing exactly who did what in the system, and when.
- To guarantee that applications can be audited.

According to commercial law, these measures (i.e. preemptive controls in the internal control system) should prevent violations of any legal restrictions on the erasure of electronically stored data. They should also guarantee legally required audit trail traceability and ensure that no violations against complete and orderly accounting occur. These measures ensure, then, that no data which is unauthorized, incomplete, incorrect, or posted to the wrong period or account is entered into the system.

2.2 Requirements

The access protection system must ensure that only authorized individuals have access to the system and to particular data. It must be possible to key in the corresponding codes (passwords) without others being able to see them.

The system should ensure that:

- only passwords of a defined minimum length are accepted,
- certain sequences of characters that could be easily guessed are not accepted,
- the password may be defined and altered by the user only,
- the system automatically demands the password to be changed at defined intervals,
- passwords are protected against being divulged to anyone other than the user him/herself.

The authorization concept must ensure that the user's rights of access are restricted to those activities in the system that are absolutely imperative for him/her based on their function/responsibility within the firm (principle of minimal authorizations); in other words, the concept must envisage as deep a hierarchical structuring as possible in respect of

- the nature of the data access (reading, creating, changing, deleting),
- programs,
- data or data files,
- functions (menus, menu lines)

in conjunction with as many different combinations of these levels as possible.

2.3 SAP facts

A new process for creating and maintaining user masters, profiles and authorizations, and for protecting access to the system, was introduced with Release 5.0. With the new authorization concept, you can allow and control user access to the R/2 system at a more precise level, and with more flexibility than in earlier versions of the software. You can now assign a user different authorizations for different company codes, i.e. the ability to change data in company code 01 but only to view data in company code 02. The new concept also contains security measures which reduce the chances of unauthorized logons or unauthorized accesses of user masters and system authorizations. Following are some facts on managing user master records and authorizations:

2.3.1 The authorization concept - basics

Authorization **objects** (i.e. data, transactions, tables etc.) are protected by **authorizations** or collective authorizations which are allocated to the objects like locks to a door. They contain **Values** (i.e. a concrete company code 01) for **fields** which you define for an accompanying authorization object (BUK). Authorization **Profiles** are like keys for the user, and a collective authorization profile is comparable to a set of keys on a key ring. These 'keys' are entered in the user's master record. A check is made to determine whether a user's authorization profile is appropriate for an authorization, similar to determining if a key fits a lock, when, for example, a program is running and the word AUTHORITY CHECK appears in the ABAP coding.

2.3.2 Collective authorizations and profiles

You can build complex authorization structures from simpler ones by combining authorizations to collective authorizations and profiles to collective profiles. With this technology, you can create a collective profile for accounts receivable clerks in company code 01, for example. Each employee with that job description will then have this collective profile in their user master. If the profile needs to be changed, you will only have to do so once, not in each user master.

2.3.3 Separating maintenance and activation

Thanks to some new transactions, the maintenance and activation of authorization components and user master records can now be separated, a development which further improves security. Also, you can now restrict a person's maintenance authorization to specific users, profiles and objects.

2.3.4 User master record

The user master record contains a listing of the profiles and/or collective profiles assigned to the user. A reverse inheritance principle applies here, meaning that changes made at a level lower than user master record level will affect all higher levels up to and including the user master record.

2.3.5 Password protection and logon

You can assign user-specific initial passwords to improve protection for passwords and logging into the system. New passwords also have to go through multiple checks.

2.3.6 Authority check

The key word AUTHORITY CHECK from the ABAP/4 programming language was reworked appropriately for the authorization check. The RAUTX macro replaces the assembler macro RAUTH for new developments.

2.3.7 Upstream security systems

In many cases companies that operate SAP systems additionally make use of an upstream security system such as RACF. In such cases it is possible to combine both systems, which as a rule spares the user the need for a double identification in the system.

If a user runs an upstream security system, this takes over the task of verifying the identity of the user (authentication). Further authorization checking takes place within the SAP System.

2.3.8 Table TSTC - SAP transaction codes

It should also be noted that authorization checks for transactions can be overridden via Table TSTC.

2.4 Risks

The high flexibility of the SAP user administration concept can lead to considerable security risks if used improperly. For example, using SAP standard transactions it is possible to close down the system, terminate the posting task and change master data without generating a log. A user with programming authorization **TM38** can remove the authorization check altogether from existing programs. Because of the complexity of the new authorization concept, many users adopt the standard profiles shipped by SAP, without really checking them, and without carefully adapting them to their own business requirements. These examples highlight the fact that security - and hence also the proper and orderly functioning - of the entire system is directly dependent on the authorizations granted.

You must, therefore, be extremely careful when allocating authorizations!

Prior to checking processing results, it is always necessary to check the user authorizations in order to make sure that the processing results are based on authorized routines and entries.

2.5 Auditing steps

2.5.1 User management

Familiarize yourself with the process for granting user authorizations (application and approval procedures, separating the 'create/maintain user', 'create/maintain objects and authorizations' and 'activate profile and authorizations' responsibilities) and also examine the internal procedures dealing with these subjects.

Find out whether written instructions exist for allocating and changing USERIDs. Obtain the relevant application forms and file them with your working papers.

Check whether official measures exist for ensuring that an employee's USERID is deleted as soon as he/she leaves the company.

Check whether the granting, changing and deleting of USERIDs has to be authorized by a responsible member of staff.

Check whether control procedures are carried out by affected departments when a new user master is created or when a user's access levels are changed.

Also check whether a user's access level is changed automatically when:

- the user's responsibilities within the organization change, necessitating changes to his/her user master record (danger of accumulated data access authority due to often-changing responsibilities in the firm),
- employees leave the firm.

Check whether employees' authorization profiles match their areas of responsibility.

Use sample testing to compare approved authorizations with authorizations actually granted.

Check whether a mandatory procedure changes profiles and authorizations if an object is changed.

2.5.2 Security and access protection

Determine whether any special security software is installed.

Have someone explain and demonstrate the logon procedures. Obtain a copy of any written documentation that is available (manuals, organizational instructions etc.).

Establish whether the system forces the user to regularly change his/her password (**TM26** or **external security software**).

Find out if the logon procedure has to be repeated whenever the system has not been used for a longer period (**TM26** or **external security software**).

Ensure that **SAP*** has been replaced in a special user master by a secret Super User.

(**Note:** If the user master record for SAP* is deleted, SAP* will revert to the original password PASS, with standard Super User privileges.)

Using transaction TMU4, display user master records and authorization information. You can analyze the following:

- User master records
- Authorization profiles
- Authorization objects
- Authorizations.

Use these analyses to determine if appropriate, job-specific authorizations were issued. Pay special attention to authorizations for employees in data processing; authorization to actually change data should only be granted for DP employees in exception cases. You should also determine what records of assigned authorizations are available.

Take note of the "**ACTIVE ON**" (= "date last accessed") field in the list of users (**TMU4**, PF16 'Misc. user data'). If this field is blank, or the date displayed is older than six months, for example, you should recommend to the SAP user being audited that he/she delete the corresponding user master records from the system.

Further manipulation can take place when these records are accessed without proper authorization. This can occur if the 'owner' of the record has never logged into the system and therefore never changed from the standard initial password. In such a case, an unauthorized user can transfer a possibly higher system authorization to his/her own identifier.

In light of this you should check to ensure that the standard initial password (also possible for all other passwords) either changes or is blocked automatically after a specific number of days. This function is defined using the 'Changing the password after X days' and 'Changing the password after X minutes' system profile parameters (SAP system audit path: Transaction **TM26** - select 'Customer profile' - 'Flow control' - parameter values: 'Changing the password after X days' and 'Changing the password after X minutes').

Check **table TSTC** to see which transactions

- are or were removed from the authorization check,
- are or were blocked,
- have been recently entered or changed.

You can see which settings are valid at any particular moment by looking in columns 'B' and 'S' in table **TSTC**:

- Column B: An X in this column indicates that users' transaction authorizations are **NOT** being checked,
- Column S: An x in this column indicates that the transaction is **BLOCKED** for all users.

You can call up the table changes log with the **RSTAPROT** ABAP program (log file) in **client 000** (also see chapter 6 "Procedure for changing tables").

Regarding table **TSTC**, you should check who can, with the following transaction authorizations,

-**TM31** 'Maintain ATAB tables' (Authorization check on/off) make changes to table TSTC in client 000. You should also determine whether a separate authorization class exists for STC table authorizations (see chap. 2.5.3 'master transactions'),

-**TM01** 'Block and unblock transactions'.

Check whether an adequate table change and release procedure exists (see chap. 6) and whether it is ensured that **all** changes to table TSTC are logged, without exception.

Find out whether it is ensured that controls are made by each individual department, for example, whether all data for which a particular department is responsible can only be altered with their authorization.

Also check that no security check objects can be switched off or deleted (i.e., table TSTC, tables with check routines, and transactions TM31, TM33 and TM38).

Check whether the procedures in place ensure that all changes are logged, so that illicit changes and/or attempts to conceal changes can be discovered.

2.5.3 Important individual authorizations

The most important authorizations should be inspected more closely. The relevant transactions in the areas of accounting and system maintenance are described below.

Note:

Transactions with mandatory posting will not only/no longer be relegated to and generated by the Accounting department alone, as determined by the increased integration of various programs (modules). This will be the case not only for documents, but for master records from other SAP systems as well.

Posting authorizations (TB01, TB02 etc.)

In SAP systems there are various transactions for posting, changing and displaying documents, not only in RF, but also in RA, RK, RM, RP and RV. In keeping with the principle of separating functions, the authorization for posting and document change transactions should be restricted to employees who are directly responsible for those functions.

Auditing steps:

Check whether authorizations for general posting transactions **TB01** and **TB02** as well as posting transactions from other areas (i.e., **PR01, TR01**, etc.) are granted only to employees of the appropriate departments who are responsible for those tasks.

Master data maintenance (TSxx, etc.)

In an internal control system (ICS), care should be taken to guarantee the separation of the 'Posting' and 'Master data maintenance' functions.

In smaller accounting departments this can be done by having the customer and vendor master data maintained by different departments, but with the responsible accounting department retaining the functions of checking and posting.

With the SAP System, you also have the option to run checks using Transactions **TS40** for master data changes to vendor accounts, **TS41** for general ledger account master data changes and **TS42** for changes to customer account master records. Another method is regular evaluation by means of the ABAP Report **RFTRAN00**, which prints and displays on screen all master data changes.

All changes to master data must be logged whenever and wherever they have the character of a document. Since it is extremely difficult to define whether or not changes have the character of a document, all changes should be included in the logging process on grounds of security.

Auditing steps:

Find out who is able to **maintain master data** (i.e. transactions TS02, 03, 05, 06, 08, 09, 40, 41, 42, PA30, 31, 32, 33, 34, 35, 61, 62, 70, 71, PW01, 03, 70, 74 etc.).

Then, determine whether these employees can also make postings.

Find out how master data changes are controlled. Check whether and ensure that the statutory retention periods are complied with.

System transactions

The system authorizations in the authorization object 'SYS' (**BI, DP, DQ, DR**) should be allocated on an extremely **restrictive** basis. You should therefore identify precisely who has these system entitlements. **System authorization DR should not be allocated in a productive system, in compliance with the ban on data erasure!**

Master transactions

The master transactions (**TMxx**) should be granted to a selected handful of users only. Exceptions to this rule are the Transactions **TM02, TM08** and **TM16**.

Auditing steps:

Find out who has authorization for Transaction **TM01** 'Block transactions'. Access to these transactions should be limited to a maximum of two members of the SAP Basis group.

Check to see who is entitled to use Transaction **TM03** 'Set system parameters'. This authorization, too, should be restricted to a maximum of two members of the SAP Basis group.

Determine who has access to the authorization maintenance transactions **TMU1/TMU2/TMU3**, and check whether these employees should be able to create and maintain SAP system users.

You should also check the distribution of **TMUx** maintenance transactions to ensure a separation of functions:

-User administrator:	Create users (TMU1)
-Authorization admin.:	Maintain authorizations (TMU2, TMU3)
-Activation admin.:	Activation(TMU6, TMU7)

Note:

- An administrator should not be authorized for the 'XU-OBJ' and 'XU-AUTH' objects. With these authorizations, a user can define, activate and assign improper access privileges.
- An activation or authorization administrator must have authorization for the 'TCD' object, in order to maintain transaction codes. With 'TCD', administrators can define and activate authorizations for transaction codes to which they themselves have no access. You should therefore check from time to time whether the authorizations defined for the transaction codes are indeed correct.

Find out who possesses authorization for the Transactions **TM17** 'Maintain number ranges', **TM21** 'Restart/Skip function', and **TM26** 'Maintain system parameters'. These authorizations should likewise be restricted to a maximum of two members of the SAP Basis group.

Determine which users are entitled to use Transaction **TM31** 'Maintain ATAB tables'. Find out which authorization categories exist for tables in table **TDDAC**, 'Allowed maintenance areas'. Investigate whether all users who have access to TM31 are merely granted access to individual categories by means of the table authorization. Each authorized user should only be able to maintain those tables that belong to his/her area of responsibility.

Find out who is authorized to use transaction **TM38** 'ABAP/4 programming'.

Because of stipulations regarding the need for "audit trail traceability", it is not permitted to use transaction TM38 in a productive system!

However, in case of emergencies a special user should be set up (for example with the name 'CRISIS') with this authorization. Every activation of this authorization must then be logged manually in a retraceable manner. In this context the "**Dual control principle**" should be observed!

Check who has an authorization to use the Transactions **VSTP** 'Stop posting task' and **VSTR** 'Start posting task'. These authorizations should be granted to a maximum of two employees.

**3 System maintenance and release procedures for
source programs and job runs**

3.1 Objective

3.1.1 Functional integrity

3.1.2 Traceability

3.2 Requirements

3.2.1 Job submission

3.2.2 Programming / Implementing the change

3.2.3 Acceptance test and production transfer

3.3 SAP Facts

3.4 Risks

3.4.1 Instability

3.4.2 Manipulation

3.5 Auditing steps

3.5.1 System review

3.5.2 Review of the procedural concept

3.5.3 Compliance with the concept

3 System maintenance and release procedures for source programs and job runs

3.1 Objective

3.1.1 Functional integrity

Changes made to programs and jobs lead to functional changes in the system. You need to ensure that only authorized changes are implemented and that all functions retain their relationships to one another.

3.1.2 Traceability

A further aim is to completely log all changes to the system to make them traceable. This is critical for changes to the source code and for changes to the Job Control Language (JCL), which generates the link between programs and datasets.

3.2 Requirements

In light of these objectives, sufficient rules must be in place for job submission, implementing the change, the acceptance and transferring to productive operation. These are as follows:

3.2.1 Job submission

Every programming change must be described in detail in a change order and must be formally approved by the 'owner' of the data. This includes both authorization for changes to programs and jobs, and the transfer of production data.

3.2.2 Programming / implementing changes

The starting basis (the object to be changed) must be secured, that is the change to the latest productive version (and not simply to a program that the programmer happens to have in the development library). Care should also be taken to ensure that only one object is changed by each person.

The development and productive systems must be separate.

Source programs (modifications) altered by the user should be stored in separate libraries (for example, . MOD.SOURCE and MOD.LOAD).

SAP documentation guidelines should be followed, particularly the SAP standards on labeling source code changes. Moreover, functional changes and, where necessary, organizational process changes should be documented.

It is also useful to define internal standards (such as for the use of file and library names).

3.2.3 Acceptance test and production transfer

The **dual control** principle should be observed for acceptance tests, that is, they should be done independently of the programmer. As a rule, they should be performed by the employee requesting the change.

More extensive program changes should be performed on the basis of a test script.

The requester of the change should, if possible, have his/her own acceptance system in which the corresponding tests can be carried out.

A source code comparison is highly recommended in order to guarantee that only those program components that are relevant to the commissioned changes have actually been modified.

You must also check whether, following acceptance, no further changes to the program are possible or that only those changes that are absolutely necessary for productive transfer (for example, changes to file names in the case of job runs) are carried out.

For this reason,

- the transfer of the altered object to productive operation (including any necessary modifications) should be automated as far as possible and be subject to mandatory logging, and
- the accepted object should be as safe from manipulation as possible and uniquely identifiable (i.e., by the conversion date).

The acceptance process must be logged. The order and release form, along with the compiler lists, should be archived (in consultation with the responsible auditor).

3.3 SAP facts

SAP supplies source and object codes (library names SOURCE and LOAD).

The **SOURCE** contains:

- Assembler programs
- Assembler macros
- Assembler dynpros (screen generation programs)

SAP naming conventions are:

- Dxyyyy: Dynpro yyy for work area xx (such as D38100)
- SAPPGxx: Main online program for work area xx (such as SAPPG38)
- Pxx...: Module for SAPPGxx (such as P38PARAM)
- SAPVBxx: main updating program for work area xx (such as SAPVB38)
- Vxx...: Module for SAPVBxx (such as V38PARAM)
- SAPxyyy: (mostly) Batch program SAPxyyy for application Rx (such as SAPF110)
- xyyy....: Module for batch program SAPxyyy (such as F110PARA)
- Qxxxx: Record layout definition xxxx (such as QKUNA)

There is a **SAP standard** for the labeling of source code changes (see SAP standards manual S24.2).

The **time stamp** of an object version can be identified through BROWSE (under TSO on IBM); in the online system with OK-Code 'D'.

SAP leaves **library administration**, especially version management, to the operating system.

But there are various **SAP processes**, for example, for assembling and linking (for details, see the library SAP.PROC).

Generated (converted) programs are usually located in the **LOAD library**, whereas converted dynpros have to be loaded into the SAP **file ADYP**.

3.4 Risks

3.4.1 Instability

The SAP System consists of different, changeable components that in their entirety are dependent on one another (complexity). For this reason, unchecked changes can easily lead to uncertainties and instabilities, such as:

- Errors may not be identified immediately.
- Data may not be processed, be processed incompletely, or be processed more than once.
- Availability of required system functions is no longer guaranteed (the system is no longer reliable).
- There may be delays in executing functions (processing security).
- Functional interrelationships may be disrupted (batch dialog).
- It may be impossible to carry out checks, which thus become ineffective.

It is quite clear that improper changes lead to considerable risks.

3.4.2 Manipulation

The complexity of the system and its vulnerability to errors caused by uncontrolled changes may lead to instabilities which can be taken advantage of.

However, in the absence of adequate and sufficient control functions, there is a fundamental danger of manipulation.

3.5 Auditing steps

3.5.1 System review

The first requirement is to get an overview of all programs and jobs (if this has not already been done in a system overview). It is important that a clear description of the function of each job is also available.

Next, perform a review of present systems (if required for the audit) and of documentation for system maintenance and release procedures for programs (source programs) and job runs.

3.5.2 Review of the procedural concept

The above-mentioned requirements should be the basis for evaluating the system concept. You should also perform a general check of the system, bearing in mind the objectives and risks detailed above. In doing so you must check that an adequate separation of functions exists.

3.5.3 Compliance with the concept

Perform both top-down (from the change order to the program) and bottom-up (from the program to the change order) sample checks of all change orders.

Since SAP jobs are usually controlled by a table (such as 102V or 902V), you should also check the authorizations for making changes to these tables.

4 A B A P

- 4.1 Objectives
 - 4.1.1 Functional integrity
 - 4.1.2 Traceability

- 4.2 Requirements

- 4.3 SAP Facts

- 4.4 Risks

- 4.5 Auditing steps
 - 4.5.1 Changing and executing ABAPs without
 changing the source code
 - 4.5.2 TM38 authorization
 - 4.5.3 ABAP authorization
 - 4.5.4 Combination TM38 authorization
 + SQL
 - 4.5.5 Generating and transferring ABAP
 programs

4 A B A P

4.1 Objective

4.1.1 Functional integrity

Changes made to ABAP programs and reports lead to functional changes in the system. You need to ensure that only authorized changes to ABAP programs and reports are implemented and that all functions retain their relationships to one another.

4.1.2 Traceability

A further objective for ABAP programs must also be that all system changes are documented completely and are therefore traceable (see ch. 3.1).

4.2 Requirements

You must check whether:

- proper functioning is guaranteed
- security is not compromised and
- programs are implemented efficiently and cost-effectively.

4.3 SAP facts

ABAP (Advanced Business Application Programming Language) is a fourth generation programming language developed by SAP. It is intended for use in an SAP environment, but can also be used in other constellations.

ABAP is an **Interpreter** programming language with language elements that contains all capabilities for:

- data selection
- sorting
- print editing
- data modification (INSERT, UPDATE, MODIFY, DELETE)
- data protection (AUTHORITY-CHECK)
- uploading/downloading (PC).

However, ABAP programs can also be compiled and then executed as a load phase from a load library.

The calling up and executing of ABAP programs is not logged. The user, however, has the option to write the corresponding instructions for logging into each ABAP.

ABAP programs which make changes to tables are not logged in all cases, unless this feature was written into the program itself.

The range of functionality for ABAP transactions **ABAP** and **TM38** can be restricted via the following authorization objects: PROGRAM, EDITOR, and SAP-SQL.

Given the authorizations for ABAP transactions (TM38, ABAP) or given rights of access to the datasets, programs can be generated and executed immediately online in the productive environment!

SAP systems are delivered with a great number of prefabricated ABAPs (standard reports). Such programs can be applied either in unchanged form or in a **modified** form.

For further details on the ABAPs and their scope of applications, please refer (inter alia) to the following SAP literature:

- ABAP/4 manual (S15.2)
- online programming in ABAP/4 (S28.2)

4.4 Risks

The risks associated with ABAPs are basically the same as for any other programming language and any other program (COBOL, PL/1 etc.).

However, allowing the TM38 editor function in a production environment entails very serious risks indeed. The editor - if it is not blocked - allows a user to get around the principle of separating the development and production environments, meaning that program development and execution can be carried out by a single person without any kind of check or approval, IN THE PRODUCTIVE SYSTEM.

If the ABAPs are not classified, i.e., not designated for specific departments and/or users, you can have situations where ABAPs from other departments are run for the wrong area, or programs could be generated, changed and executed improperly (via TM38).

4.5 Auditing steps

4.5.1 Changing and executing ABAPs without changing the source code

An ABAP report can be edited - and thus potentially manipulated - online. You can then directly execute a program using SUBMIT without the need for a prior restore (UPDATE) which would normally guarantee a recordable change in the source.

ABAP evaluations that are requested by the auditor for control purposes should, for security reasons, always be run as a batch job; care should be taken here to ensure that the cover sheet is not removed (option). Make sure you obtain the list together with the control information (job control, ABAP selection criteria). This is the only way of ensuring whether the productive system (productive databases) and the selections have been processed in accordance with the auditor's instructions.

General rule: Never trust an ABAP that has no cover sheet!

4.5.2 TM38 authorization

TM38 is a transaction for the maintenance and execution of ABAP programs. One of its important functions is the Editor. Execution in batch mode presupposes that access authorization exists to the addressed databases. In the online mode this authorization is already implicitly granted by means of the general SAP System authorization (IMS).

TM38 authorization should not exist in the productive system!

The SAP System contains what are known as request-ABAPs, which start other ABAPs in the form of a procedure and usually carry parameters that require updating (company code, date etc.). The maintenance of such special parameter-ABAPs likewise requires TM38 or - which is more advisable - an organizational solution using parameter tables, the handling of which should be integrated into the authorization concept.

As of Release 5.0, the system offers the capability of working with report variants. Reports of this kind are assigned to a separate report category for which authorizations need to be allocated based on requirements. The ability to make changes is then confined to the preset variables in the report.

Auditing steps

Use transaction **TMU4** to determine:

- whether **TM38 authorizations** are granted in the productive system.
- to what extent program group names are allocated to the authorization objects PROGRAM and EDITOR with each P-GROUP field.
- if maintaining attributes and text elements is possible (PROGRAM/P_ACTION: EDIT),
- if editing ABAPs is possible (EDITOR/EDT_Action: EDIT or VARIANT) and
- if a data-changing SQL access is possible (SAP-SQL/SQL_ACTION: SAP-SQL).

If so, find out in each individual instance why these authorizations were granted or still exist, and demand the written instruction for creating these authorization(s).

4.5.3 ABAP authorizations

With ABAP authorization, users receive access rights for individual authorization groups in which ABAP programs can be grouped for the purpose of program execution.

The information provided up to this point for transaction TM38 also generally applies for the ABAP transaction code, with the difference that no source editor functions are available.

Auditing steps

Check which users have access to which ABAP authorization groups. Check these groups and the granted authorizations. Check whether the authorizations exist in written form and are still required.

4.5.4 Combination TM38 authorization + SQL

With this combination programs with data-changing accesses can be written and executed.

Auditing steps

Make sure that no one has this authorization in the productive system.

4.5.5 The generation and transfer of ABAP programs

There is basically no difference between ABAP and any other programming language in this matter. For further details, refer to the chapter entitled 'System maintenance: sources, jobs, runs and procedures'.

5 Computer center job submission procedure, job documentation and computer center logs

5.1 Objective

5.1.1 Job submission procedure

5.1.2 Job documentation

5.1.3 Computer center logs

5.2 Requirements

5.2.1 Job submission procedure

5.2.2 Job documentation

5.2.3 Computer center logs

5.3 SAP facts

5.3.1 Batch programs

5.3.2 Online programs

5.3.3 Logs

5.4 Risks

5.5 Auditing steps

5.5.1 Review of existing procedures

5.5.2 Review of procedural concepts

5.5.3 Checking for compliance with stipulated procedures

5.6 SAP Job documentation (sample proposal)

5.6.1 General

5.6.2 Prerequisites for the job start

5.6.3 Required post-processing after job run

5.6.4 Restart measures

5 Computer center job submission procedure, job documentation and computer center logs

5.1 Objective

5.1.1 Computer center job submission procedure

The following are the primary objectives of the procedure:

- Protection of corporate and personal data
- Data and functional integrity
- Resource protection

5.1.2 Job documentation

Goals of the job documentation are:

- Ensure an error-free job run
- Ensure independence from individual persons' knowledge
- Technical data processing functions must be inspectable, by an outside expert, in a reasonable period of time.

5.1.3 Computer center logs

These are necessary to demonstrate orderly processing, particularly compliance with the computer center job submission procedure.

5.2 Requirements

5.2.1 Computer center job submission procedure

The computer center job submission procedure must clearly define processes and responsibilities regarding

- order placing
- implementation
- post-processing and output distribution.

No job may be executed without an order; as a rule the order will be placed by the department concerned.

5.2.2 Job documentation

The job documentation contains:

- Console commands
- Starting and exiting the online system
- Description of all jobs according to a uniform pattern(see sample proposal)

Documentation must be retained for 10 years!

5.2.3 Computer center logs

Computer logs show when which job was run and with which control statements (for a certain time). They have mandatory retention periods as needed for traceability.

Exception reporting in sensitive areas must be based on computer center logs.

5.3 SAP facts

5.3.1 Batch programs

SAP provides a variety of batch programs (such as the payment program **SAPF110** and the document reorganization program **SAPB80U**). These can be found in the SOURCE and LOAD libraries.

You can use the '**JOBS**' transaction to generate JCL. You must first define required information, such as job card, file name, etc., in a job profile. Use the **RSTRDOKU** or **RJTRDOKU** reports to generate and print documentation for the JOBS transaction. All jobs generated should be copied from the generating library to another library, otherwise any changes would be overwritten if another job was then newly generated.

All SAP batch programs run under the SAP batch main program **SAPBTCH**; this is a standardized, operating system-independent environment for batch programs with a variety of utility and test functions.

Once **SAPBTCH** has been called up, only SAP's own control statements are valid.

You can find these control statements either directly in the job stream or you can take them from an SAP table by means of a special statement (for example, **\$INPUT TAB T102V, F110**).

SAP supplies sample control statements in Table **102V**.

The documentation for the programs is located in the system and can be called up using Transaction **DOKM**.

5.3.2 Online programs

Diverse tasks must be started for the SAP online system; the most important ones are:

- the dialog task **SAPDISA**
- the update task **SAPVISA**
- the spool task **SAPSISA**

5.3.3 Logs

The following SAP-specific logs are available:

- Logs of batch programs
- SAP log file; this can be read using ABAP programs **RSSLOG00, RSBLOG00** or with the '**SLOG**' transaction.

5.4 Risks

1. Unauthorized (read) access to corporate and personal data
2. Unauthorized, uncontrolled, unnoticed changes to data and programs
3. High resource load due to inappropriate program compilation
4. Improper operation, especially in exception cases (error processing)
5. Dependence on the knowledge of individual persons.

5.5 Auditing steps

5.5.1 Review of existing procedures

The existing procedures for

- computer center job submissions
- job documentation
- generating and handling computer center logs

should be recorded and documented (to the extent required for auditing).

5.5.2 Review of procedural concepts

The stipulated procedures should be checked against the requirements and risks detailed above.

5.5.3 Checking for compliance with the stipulated procedures

Checking the compliance with the procedures is carried out

- by inspecting the documents
- by evaluating log files and logs.

5.6 SAP job documentation (Sample proposal)

The following is a sample proposal for a typical SAP job documentation format, which should not, however, be considered exhaustive.

5.6.1 General

Responsible for the content:

Job identification:

Call-up:

Brief description of task:

Frequency (on request, daily, weekly, and so on):

Estimated run-time:

Whether processing possible parallel to online operation:

Particulars of job:

Programs within the job

5.6.2 Prerequisites for the job start:

Required peripherals:

Security measures prior to job run:

Which tables/Control instructions are needed:

Other prerequisites for the job start:

5.6.3 Required post-processing after job run:

Required data carriers/forms for output:

Update run required after job:

Follow-up jobs:

Security measures after job run:

5.6.4 Measures needed in restart cases:

Error can be corrected directly:

- Correct error
- Reload the secured data
- Restart the job

Error cannot be corrected directly:

- Reload the secured data
- Generate error log
- Inform the responsible person

Job name for data reset:

Which jobs may no longer be run following abnormal system termination:

6 Procedure for changing tables

6.1 Objective

6.2 Requirements

6.2.1 Logging

6.2.2 Customer tables

6.2.3 Access protection

6.2.4 Job/Organizational instructions

6.2.5 Safeguarding the information flow

6.3 SAP facts

6.3.1 Purpose and structure

6.3.2 Table access and logging

6.3.3 Area of validity and customer tables

6.3.4 Important ABAP reports/programs

6.4 Risks

6.5 Auditing steps

6 Procedures for changing tables

6.1 Objective

The goal of a procedure for making changes to tables is to safeguard the correct table setting, and to ensure the retraceability of all change steps.

6.2 Requirements

6.2.1 Logging

All (relevant) table changes should be logged. The following programs are available for this purpose:

RSTAPROT 'table changes(log file)' for day-specific online evaluations.

SAPACCU 'ATAB change descriptions (batch)'

RSACCU00 'ATAB table changes from log file (batch)'

The **SAPACCU** and **RSACCU00** programs necessitate one another. These programs allow you to evaluate batch table changes and change histories. You can find further information on these in the program documentation.

You should guarantee a **daily logging** of all table changes. The lists should be automatically summarized, to the monthly or quarterly level, for example. Random sample checks should also be made of the logs in the case of 'critical' tables.

The retention period for all logs is 10 years (in Germany, for example).

6.2.2 Customer tables

You must adhere to the SAP naming convention (9 series number range).

The documentation also needs to be adequate.

6.2.3 Access protection

There must be an authorization concept to guarantee access protection. This must regulate, among other things,

- which authorization class is intended for which purpose or for whom
- how the special protective measures that may be defined in the table header are to be used
- how client references are to be used.

The object **RSTABLE** allows the user to flexibly define table-specific authorizations when setting up profiles.

6.2.4 Job/organizational instructions

To ensure that the organization does not rely solely on the knowledge of individual persons, and to ensure security for a correct table setting, "job/organizational instructions" should be available for certain tables in which, for example, the following information may be filed:

- naming conventions
- consequences ensuing from table changes
- causes of a table change
- reason for the current setting.

6.2.5 Safeguarding the information flow

Because of the SAP system's high degree of integration, making a change to a table may lead to unintended side effects (in other modules, for example). For this reason there should be a binding procedure which, in the case of "critical" tables, safeguards the flow of information to the people and/or departments affected by a table change.

6.3 SAP facts

6.3.1 Purpose and structure

Tables are intended to enable company-specific adjustment of the standard software, without requiring modification of the programs. They contain variable factors influencing requirements for:

- flow control
- logical checks
- calculation routines
- automatic generations
- screen modifications

Tables consist of:

- an argument section
- a functional section
- a table header control block
- max. 4-digit table names (for example T001, 001T, RDIR)

The **table header** contains details on such things as:

- generation and last access
- authorization classes for TM08 and TM31 access.

Tables are stored in the file **ATAB**, and the table header in the Data Dictionary (file **DLIB**)

There are two **types of tables**:

- edited tables
- non-edited tables

6.3.2 Table access and logging

Table access is possible with

- the online transactions TM08/OM08, TM31/OM31, TM33, DDIC
- the batch program SAPTABU
- ABAP

The online transactions may be broken down as follows:

- TM08: Display table content
- OM08: for non-edited tables
- TM31: Maintain table content
- OM31: for non-edited tables
- TM33: Maintain table attributes; restricted functionality (see under DDIC)
- DDIC: Maintain table attributes via PF20 in the Data Dictionary

SAPTABU

SAPTABU is a comprehensive utility which contains, for example, the following functions:

- maintaining table entries
- table comparisons (for instance, test system versus productive system)

The documentation can be called up using Transaction DOKM

ABAP

By means of the ABAP instruction MODIFY, both ATAB and ABAP-internal tables may be changed)

Logging

The use of TM31 is logged as a standard feature in the file APLZ (after the update has been effected directly via the update task!): a daily evaluation is possible by means of ABAP **RSTAPROT**.

Generation and last access are logged in the table header. ABAP: the user must him/herself ensure logging in the APLZ file.

6.3.3 Area of validity and customer-specific tables

Tables are basically client-specific; however,

- In a new client, C2, you can avoid creating separate tables and instead make cross-references to already existing tables in another client C1 (TM33). The table can only be maintained in the original client, C1.
- Some system tables are only available in client 000 and for all clients (for example, 000 client table, RDIR Directory table for the ABAP reports).
- Some tables are client-independent (T022B, for example).

Customer-specific tables should, according to the SAP convention, begin with 9. Use transaction '**DDIC**' to make this definition.

6.3.4 Important ABAP reports

Important **ABAP** reports for tables are the following:

RSDOKTAB: table manual
RSTAPROT: log of table changes
RSSCAN00: search a table for a text string
RSACCU00: ATAB table changes from log file (batch)
SAPACCU: ATAB change descriptions (batch).

NOTE:

All authorization checks are excluded for the RSSCAN00 program. Sensitive data should therefore not be stored in tables!

6.4 Risks

Since tables in the SAP System take on a central control function, the risks of an inadequate procedure for altering the content of tables are quite obvious:

- There is a danger of incorrect settings.
- Making changes to a table may trigger unintended and unwanted consequences elsewhere.
- This can disrupt the integrity of the dataset.

It is also conceivable that authorizations may be changed, if these are defined in tables (i.e., release authorizations for parked documents).

6.5 Auditing steps

In general,

- you should determine the procedure for altering tables,
- it should be evaluated based on the requirements listed in the preceding section, and
- compliance should be reviewed by sample checking.

Some examples of possible audit actions are detailed below:

- Establish which tables are relevant, for example, for application 'X' (entry of '*X' in the table name in **TM08** or by using **SAPTABU**).
- Generate a list of tables for each authorization class (batch program **SAPTABU**).
- Manually go through the list of all tables in the system to check compliance with the naming convention (generate list with **SAPTABU**).
- Do sample checks on the customer-specific tables documentation (**RSOKTAB** with table name **T9** or **T9***).

7 Batch input interfaces

7.1 Objective

7.2 Requirements

7.3 SAP facts

7.4 Risks

7.5 Auditing steps

7 Batch input interfaces

7.1 Objective

According to most legal standards, data must be recorded, stored and processed in its entirety, correctly, and in a timely and orderly fashion. It may not be falsified in the event of changes. Therefore, there should be controls during these data processing activities to ensure compliance.

7.2 Requirements

These preconditions must be met to allow for effective controls:

- Organized work flows
- Separation of incompatible functions
- Control measures and entities.

The organization of the work flows heavily influences the effectiveness of the internal control system. Work flows should therefore be designed so that any deviations from standard procedure will be detected and recorded by controlling entities.

An effective internal control system requires the separation of certain functions. A distinction between functions for planning, executing and monitoring should be drawn.

7.3 SAP facts

A standardized batch input interface transfers posting data from external source systems to the RF system. It is important for the user to generate records in a format which corresponds to the structure of the required screen masks.

The data are stored in a file (session) and posted online from there. The online processor processes these data by merging them. The same program processes and check routines are run, as would be the case with manual entry. After processing is completed for a session of batch input transactions, the session will be given an identifier with the status "successfully processed" (FIN) or erroneous (ERR). The processing clerk can carry out the error post-processing on screen with the manual entry correction tools which are also available. This method does not require batch to be repeated; instead, all that needs to be done is to post-process the faulty part of the erroneous transactions.

As an alternative to processing by the particular department, you can also start session updates via the console.

The batch input interface also executes the transfer of master records and open items.

You can validate batch input session names via the BTC authorization object.

Without the value 'BI' in authorization object 'SYS', batch input sessions cannot be processed.

7.4 Risks

There is a danger with non-SAP source systems that the data validation checks may be handled differently than the SAP tables. This may affect master and transaction data.

If the transfer procedure is inadequately regulated, this may lead to inconsistent accounting by period between general ledger and sub-ledger accounting.

If session names cannot be validated, employees with batch input authorization may be able to start, correct and delete sessions from other departments (depending on their transaction code authorizations).

7.5 Auditing steps

Due to these control requirements, the auditor should ask the following questions:

Is there an overview of all interfaces to SAP including such specifications as the

- transferring work area
- data content
- file name
- period
- session name
- processing job
- relevant tables
- reconciliation group
- responsibilities?

Which users are allowed to generate, start, correct and delete sessions? (Transaction TMU4 'Information system for users and authorizations')

Is there an overview showing which session names are reserved for which department?

Who reconciles the posting data of the processed sessions?

Who checks to see that the data from the source systems are transferred in their entirety, correctly, and in a timely fashion?

8 Master data changes

8.1 Separation of functions

8.1.1 Objective

8.1.2 Requirements

8.1.3 SAP facts

8.1.4 Risks

8.1.5 Auditing steps

8.2 Traceability

8.2.1 Objective

8.2.2 Requirements

8.2.3 SAP facts

8.2.4 Risks

8.2.5 Auditing steps

8 Master data changes

8.1 Separation of functions

8.1.1 Objective

Depending on the size of a company, a separation of functions is necessary to achieve sufficient security, and also to provide internal control system for guaranteeing compliance with generally accepted accounting principles.

This is not as critical for small companies which, for organizational reasons, cannot achieve separation of functions. In this case special - individually designed - checks on master data changes are necessary in order to achieve a sufficient standard of security. For larger companies, however, such checks cannot take the place of an ongoing separation of functions.

8.1.2 Requirements

Responsibilities/authorizations for master data management need to be clearly defined.

An appropriate separation of functions between the EDP and accounting departments on one hand, and between master data maintenance and posting activity on the other.

8.1.3 SAP facts

The SAP System's design and comprehensive, differentiated spectrum of authorizations promote an adequate separation of functions.

8.1.4 Risks

An insufficient internal control system impairs both accounting according to accepted principles and data security. This applies particularly for payment transaction security.

8.1.5 Auditing steps

Who has the authorization for creating and changing customer, vendor and general ledger master data?

For larger departments, are each of the authorizations for access to their customer, vendor or general ledger accounts restricted further?

8.2 Traceability

8.2.1 Objective

Traceability in accounting must be guaranteed at all times. The principles of "posted is posted" and the "erasure ban" in accordance with generally accepted accounting principles and commercial law must be ensured.

8.2.2 Requirements

Adequate logging, documentation and retention of master data changes and deletions.

Background: record posting data in keeping with generally accepted accounting principles for the required data retention period.

8.2.3 SAP facts

A complete recording and traceability of the posting data are basically ensured. However, some authorizations exist which enable users to change a dataset without logging the transaction (see chapter 2 'Security and access protection').

8.2.4 Risks

Jeopardization of complete posting data recording and jeopardization of the proper and orderly accounting principle could result.

8.2.5 Auditing steps

Have **authorizations** been granted that allow users to change the dataset without logging?

Are the **changes** to master data sufficiently traceable?

Are master data **reconciliation logs** printed out and checked by the supervisor?

Check the **change documentation** of master data on the basis of the following **ABAPs**:

- **RFTRAN00** Documentation of master data changes
- **RFTRAN05** Master record changes - log file
- **RSAEBELG** Master data change documents
- **RSTS26D** Master record changes: business area
- **RSTS40D** Master record changes: vendor master
- **RSTS41D** Changes general ledger account master data
- **RSTS42D** Customer master data changes
- **RMTRAN00** Material master and info record changes
- **RSTX09D** Material master data changes
- **RMPRAE00** Materials price changes
- **RMTR2120** Price changes via batch input
- **RMWIPR00** Display transactions for document.

9 Posting data reconciliation

9.1 Objective

9.2 Requirements

9.3 SAP facts

9.4 Risks

9.5 Auditing steps

9 Posting data reconciliation

9.1 Objective

According to generally accepted accounting principles, business entities are required to keep books and, in doing so, comply with accounting rules and regulations. If accounting tasks are carried out by means of data processing systems, the company must ensure that these data processing systems comply with accounting rules and regulations.

This means that the head of the department or the person responsible for posting data has an obligation to monitor processing results. The aim of these controls is to ensure that the data is:

- complete
- genuine
- correctly evaluated
- posted to the right period
- allocated to the proper accounts
- totaled correctly
- and has a correct final posting.

9.2 Requirements

Data completeness must be ensured, regardless of the degree of integration of the SAP applications. This means that users should be able to reconcile processed posting data at all times. SAP provides processing controls to help you achieve this level of data completeness, which are installed at various levels of the application. Preventive and detecting controls must supplement one another here.

9.3 SAP facts

Original documents are entered in document sessions. Each session is given control totals, such as the number of documents, debit/customers total etc. The RF system carries a control total memory per user (**TB21**) to enable a comparison with the specified control total.

Document storage guarantees that the balance of all postings remains zero. This can be verified at any time using the balance sheet/profit and loss reporting.

The document number is uniquely assigned to one document. If document numbers are automatically assigned by the system (File **SYSV**), the documents are numbered consecutively; the system checks for duplicate number assignment in the case of externally allocated numbers.

The documentation relating to all documents within a selected period is located in the document compact journal **RFBELJ00**. The summary report of **RFBELJ00** shows the reconciliation summary sheets. These reconciliation totals have to be manually transferred to a list, and then accumulated with the amounts carried forward.

In order to reconcile the accounting figures, the "**accounting reconciliation**" action needs to be performed. It ensures that documents exist to support all monthly debits and credits.

Reconciling G/L account postings

G/L accounts are reconciled based on the balance audit trail. The lists from **RFHABU00** 'G/L from residence' or **RFKLBU10** 'G/L from historical balance audit trail' should be reconciled with the G/L account balance list **RFSSLD00**.

Reconciling the open item accounts

Open item accounts are reconciled by comparing the customer and vendor balance lists **RFDSL00** and **RFKSLD00** with the lists from the open item account balance or historical balance audit trail.

- Open item account balance audit trail:
RFKKBU00 or **RFKKBU10**.
- Historical balance audit trail: **RFKLBU10**.

For **short-term** reconciliations (i.e., at monthly level) of the C segments of the customer/vendor/G/L accounts with the debit/credit totals of the posted documents in the corresponding time period, the **RFBUAB00** program 'Reconciliation of documents/Monthly debits and credits master' can also be used.

9.4 Risks

If posting data is not reconciled on a timely basis, there is a risk that errors could occur and remain undetected for extended periods of time. The obvious consequence of this could be that management would have a faulty decision-making basis, which could then lead to serious errors.

9.5 Auditing steps

On the basis of these control requirements the following questions arise for the auditor:

At what intervals and by whom are the postings reviewed?

Is the accounting reconciliation carried out regularly?

Were any discrepancies found?

Has it been guaranteed (in the case of a release upgrade) that the closing balance lists from old releases correspond to the opening balance lists from the new release versions, and that they are correspondingly documented?

Is there any record of abnormal system terminations?

Checking **selection criteria** for the ABAPs (accounting reconciliation action)

- RFBELJ00
- RFDSL00
- RFKSL00
- RFSSL00

A check should be made here to ensure that an identical CPU date selection criteria has been made.

10 Day-end/Month-end closing

10.1 Objective

10.2 Requirements

10.3 SAP facts

10.4 Risks

10.5 Auditing steps

10 Day-end/month-end closing

10.1 Objective

An accounting system is deemed to be efficient and well-ordered if its design allows a knowledgeable outsider to obtain an overview of the business transactions and the company's position within a reasonable period of time. It should also be possible to follow the origin and processing of these business transactions.

10.2 Requirements

The basic requirements of generally accepted accounting principles demand that all business transactions subject to recording must be recorded fully, correctly, and in a timely and orderly fashion. In an integrated system, these requirements are not limited merely to posting-related transactions. A system's controlling tables, data security and data backup must also be included in these requirements.

10.3 SAP facts

Day-end closing

After the day's online operations have been terminated, the person responsible for the posting data must ensure that all data records in the **log file (APLZ)** have been properly posted (Transaction **TM06**). Prior to securing the SAP system data, it is necessary to run all reports which access data from the APLZ log file. All changes in accounting-related tables must be logged. The ABAP report **RSTAPROT** is used for this purpose. The log must be kept for the legally stipulated retention period.

The posting data should be reconciled as described in chapter 9.3.

After finishing the checks on the log file, it should be saved using the SAP standard program **SAPLOGU**. The log generated by SAPLOGU has to be retained in keeping with legal requirements.

Finally, the SAP files, including the APLZ file, should be saved using the operating system.

Month-end closing

At first, the workflow for the month-end closing contains the same procedures as in the day-end closing. There are also some additional tasks to execute, which reconcile and document the posting data.

The following steps must be completed to perform a month-end closing:

- Transaction **TB20** is used to remove the closed month from the time period in which items can still be posted.
- The required evaluations used for documenting the posting data are generated.
- The accounting reconciliation is performed.

RFBELJ00 has the function of compact journal.

The **RFHABU00** program documents G/L account postings.

The **RFHABU10** program takes the role of the general ledger for cumulated G/L account posting.

Reports **RFKKBU00** and **RFKKBU10** document the postings made over a certain time period to customer, vendor and G/L accounts.

Accounting periods

The allowed accounting periods can be displayed and maintained with transactions **TB19** and **TB20**, respectively. Two allowed (open) posting intervals can be defined.

The fiscal year includes a settlement period of up to 16 accounting periods. If the number of normal accounting periods is greater than 12, the number of special periods is reduced.

10.4 Risks

In addition to the risk of error which is inherent in non-timely reconciliation, month-end closing also carries other risks which can be eliminated using cross-system controls.

For example, an imprecise period definition can lead to inconsistencies between the general and sub-ledgers.

Subsequent evaluations for taxation or sales statistics purposes would then be based on faulty data. If the system for securing the data is inadequately organized, this hinders the task of reconstructing the original data in the event of errors.

10.5 Auditing steps

The following questions may be of importance for assessing the checks and controls:

Who is responsible for carrying out the month-end closing?

Do the rules for carrying out the month-end closing satisfy the separation of functions requirement between the functional departments and the systems department?

What measures guarantee that standard process flows will be carried out in the proper order?

Are there any rules governing error management?

Are the user profiles set up with due consideration for the separation of functions (**Transaction TMU4** 'Infosystem users and authorizations')?

Are all documents generated which are subject to a legal retention period?

11 Year-end closing

11.1 Objective

11.2 Requirements

11.3 SAP facts

11.4 Risks

11.5 Auditing steps

11 Year-end closing

11.1 Objective

As a rule, the basic functions of an accounting system comprise the following ledgers and documents:

- opening balance sheet
- document collection
- compact journal
- general ledger
- inventory
- balance sheet with profit and loss statement.

These basic functions are indispensable components of any bookkeeping system. They must incorporate the checks and controls that monitor the following performance criteria:

- completeness
- genuineness
- correct valuation
- assignment to the correct period
- assignment to the right accounts
- correct summation
- correct final posting.

11.2 Requirements

Legislation may require that automatically processed or stored accounting systems must contain document collection, compact journal and general ledger, and that such data may be subject to retention periods of (in Germany, for example) six to ten years, whether the posting voucher is actually printed out or not.

The controls for day-end and month-end closing must also be included in the procedure for year-end closing. In addition, controls for non-recurrent tasks are also required. For these tasks in particular, responsibilities must be clearly regulated.

Transaction data has to be documented. This can be in paper form or by filming of the journal and the accounts.

11.3 SAP facts

The balances carried forward for creditors, vendors and balance sheet accounts are automatically adjusted for postings to the prior year. Profit and loss accounts are carried forward to a retained earnings account (**Table T030**).

Postings may be made to the previous year and to the current year in parallel. After year-end closing, the previous year has to be blocked using a table entry.

Balance sheet and profit and loss statement

The balance sheet and profit and loss statement are printed out exclusively by means of ABAP. The **RFBILA00** report is available as a standard feature.

The list contains financial statements (balance sheet and income statement) for a freely definable period within a given fiscal year with absolute and relative comparisons to a comparison period.

You can generate four financial statements with this report, structured according to different classification principles. The classification principle and output screen can be controlled by various **tables (T011, T012, T013, T014)**. You define the balance sheet as well as the income statement within a table. The first digit of the financial statement key determines the account allocation in the balance sheet and income statement:

- 0 = non-assigned accounts
- 1 = balance sheet (assets side)
- 2 = balance sheet (liabilities side)
- 3-9 = income statement.

Balance carried forward

The balances for the KUNA, LIFA and SKSA files have to be carried forward in separate runs using the **SAPB011** program.

Reorganization/Archiving

Fiscal years that are no longer required are to be archived using programs **SAPB01U** and **SAPB02L**.

11.4 Risks

Risks can arise in connection with the balance sheet and income statement due to the nature of the accounts summarized in the year-end closing item. If the allocations do not correspond to formal requirements, the year-end closing can be rejected. If data is incomplete, incorrect information could be transmitted to management.

Non-compliance with the classification principles and statutory retention periods may have legal consequences.

11.5 Auditing steps

The same basic questions need to be researched as in a month-end closing; however, these are supplemented by some specific problems of year-end closing which relate to the balance sheet display, posting data documentation and the documents that are subject to retention periods.

The user profiles issued (**Transaction TMU4** 'User and Authorizations Info System') must show the responsibilities for year-end closing clearly and unequivocally.