



COVER SHEET

Best, Peter (2000) Auditing SAP R/3 – Control Risk Assessment.
Australian Accounting Review X(3):pp. 31-42.

Accessed from <http://eprints.qut.edu.au>

Copyright 2000 CPA Australia

Auditing SAP R/3 – Control Risk Assessment

Peter J Best

Associate Professor
School of Accountancy
Queensland University of Technology
Brisbane Australia
Email: p.best@qut.edu.au

Abstract

This paper provides an introduction to auditing in an SAP R/3 environment, focusing primarily on the assessment of control risk. A number of distinguishing characteristics of the SAP R/3 system which affect the audit are described. The application of a standard internal control framework to the assessment of application controls is illustrated. Two significant pervasive general control areas are examined – system development and program maintenance, and user access control. Relevant controls in these areas are discussed. Methods for auditing these controls are outlined. Several opportunities for research in the auditing of SAP R/3 are introduced.

1.0 INTRODUCTION

SAP R/3 belongs to the family of *enterprise resource planning* (ERP) systems. An ERP system has several distinctive characteristics [Norris et. al., (1998)]:

- Multi-functional in scope – it tracks financial results (dollars), procurement (material), sales (people and goods) and manufacturing (people and resources);
- Integrated in nature, that is, when a piece of data is entered regarding one of the functions, data regarding the other functions is changed;
- Modular in structure, that is, it can be used in a way that is as expansive or narrow as you choose.

Organisations may develop an ERP system through the acquisition and integration of ‘best practice’ specialist software (eg. financials and payroll) and internally developed modules (eg. equipment hiring), thereby producing a ‘multi-vendor’ solution. SAP R/3 is a ‘single-vendor’, packaged ERP system. R/3 is generally considered as the ERP system with the broadest and deepest available functionality, but has also the ability to integrate with non-SAP systems [Norris et. al., (1998)].

ERP systems present challenges to auditors not only because of their breadth, but because of their technical complexity. The client may have implemented the system using a variety of hardware and software platforms. Multi-vendor ERP solutions introduce an additional complexity – the interfaces between the various systems, eg. exporting data from the payroll system for import into the financial accounting system.

This paper focuses on the auditing of SAP R/3. This focus is justified given R/3’s dominance of the ERP market [Gilbert (2000)]. R/3 is a popular system due to its functionality and flexibility. Its ability to be customised to suit an organisation also contributes to its complexity. This paper introduces some of R/3’s characteristics and outlines control risk assessment in such an environment. An internal control framework for assessing application controls is illustrated. The assessment of pervasive general controls is examined. Several opportunities for further research are also outlined.

2.0 OVERVIEW OF SAP R/3

This section provides an introduction to SAP R/3, providing an overview of some of its characteristics and idiosyncrasies which have an impact on the audit.

SAP, the company developing and marketing R/3, is based in Walldorf, Germany. The original company name was “Systemanalyse und Programmentwicklung.” SAP now stands for “Systems, Applications, and Products in Data Processing.” R/3’s predecessor, R/2, was developed for the mainframe environment. The R/3 system was developed to run in a client-server environment on local area networks.

R/3's popularity among medium to large organisations has been due to several characteristics [Ernst & Young (1995)]:

- SAP's modular structure allows companies to purchase the modules that they require and to link them together in a way that provides the required level of flexibility.
- SAP's highly integrated nature reduces the need for reconciliations. The flexibility of the database allows users to quickly locate data at various levels and across modules.
- The ability to expand the system in the future by adding further modules allows companies to take a phased approach to implementing SAP.
- The ability to easily maintain the system through transactions, tables and master files adjustments.
- SAP's ability to work with a number of different currencies and consolidate the results of foreign companies.
- The strong security features that may be set up to control access to transactions and data.
- The comprehensive audit trails that are produced by the standard system as well as the ability to access this information on-line.
- The machine independence of SAP allows users to easily change to a new hardware or system software environment that is supported by SAP.
- The reassurance that comes with using a system that has been comprehensively tested and used by many other organisations.
- The opportunity to benefit from future system improvements.

SAP R/3's standard modules are: FI - Financial Accounting, CO – Controlling, AM - Fixed Assets Management, PS - Project Management System, WF - Workflow Management, IS - Industry Solutions, HR - Human Resources, PM - Plant Maintenance, QM - Quality Management, PP - Production Planning, MM - Materials Management, SD - Sales & Distribution. In addition, the SAP Basis module is a mandatory, underlying component which provides the connection between these modules, standard mechanisms for user dialogue, tools for system maintenance and enhancement, and the ability to implement security.

These standard R/3 modules can support the basic business processes of most organisations. The expanding range of Industry Solutions are enhancements to the R/3 system, which have been developed to suit the needs of particular industries or types of organisation. Examples of such industry solutions are: aerospace and defence, automotive industry, banking, high-tech and electronics, real estate, chemical, consumer products, education, healthcare, metal, construction, oil, pharmaceuticals, public sector, retail, transportation and utilities [ASAP (1997)].

SAP R/3's three-tier client/server architecture distinguishes between the user interface (or "presentation"), the applications, and the database. Tier One, the user interface, appears at the workstation and provides a user-friendly mechanism for users to interact with R/3, entering transactions and displaying balances and line items. Tier Two, the applications, may be spread among various servers on a network. Tier Three, the database, may be implemented using Oracle or some other product. The applications pass data to and from

the database as they perform their functions. The R/3 system can be configured to suit the size of the organisation by allocating different processing systems to database management, application processing and presentation services. The various servers can be running any combination of operating systems, for example: presentation - Windows 3.1, UNIX, Windows NT, OS/2 and Windows 98; applications - Windows NT, UNIX, Open VMS and MPE/iX; and database - Windows NT, UNIX, Open VMS and MPE/iX. R/3 may be implemented with Oracle, Informix, and other database systems.

SAP R/3 has its own in-built program development environment. SAP software is written in ABAP/4, which is SAP's fourth-generation Advanced Business Programming Language. The ABAP/4 Development Workbench is a complete environment for creating and maintaining programs in this language. The strict control disciplines exercised by the Development Workbench ensure that resulting software will run immediately on all types of computer and all database systems supported by SAP.

All development objects are stored in the R/3 Repository. The following types of objects may be under development: dictionary data, ABAP/4 programs, dynpros (interactive programs controlling data entry and validation), documentation and help texts. Development objects are assigned to programmers and to a development class. Old versions of objects are retained and changes are logged to permit reconstruction of earlier versions. R/3 facilitates the establishment of a system landscape that separates development, quality assurance and production environments, and utilises a Change and Transport System (CTS) to control the transfer of changes in development objects from development to testing, and to production [Metzger & Roehrs (2000)].

The functionality of R/3's modules is claimed to be suitable for most organisations, being based on global 'best practice'. However, extensive configuration is still required to customise R/3 to the organisation. For example, R/3 provides standard organisational entities suitable for external reporting (eg. company code, business area) and internal reporting (eg. controlling area, cost centre, profit centre). The organisation must decide how these organisational entities should be utilised to meet management's reporting requirements. Customising R/3 includes such activities as: creating company codes and business areas, identifying the country of operation, local currency and language; defining the fiscal year and posting periods; creating charts of accounts and financial statement versions; creating general ledger master records, and sub-ledger master records; defining terms of payment for customers; and creating tax codes for Goods and Services Tax (GST) purposes. R/3 provides software, the Implementation Management Guide (IMG), to guide the implementation team through this customisation process. The IMG lists all activities for implementing each R/3 module and assists in controlling implementation. Accessing the IMG is also a means of understanding the nature of each customisation step through its on-line documentation and reviewing the customising settings.

SAP R/3 support three modes of processing – on-line interactive, background, batch input. With on-line interactive processing, the user enters transaction data or interacts with the system using its user interface. Any warning or error messages are displayed on

the screen. With background processing, the program executes without interactive user input. In batch processing, a number of transactions are entered into the system in a batch. Batch input can take place in the background where no changes to data can be made during processing or in the foreground where errors may be corrected interactively. Batch input may be used for efficiently entering large quantities of data into the system from an external, non-SAP system.

In R/3, end-users and ‘system’ users access the system through the same authentication process requiring the entry of a client identification, user name, password and language. These users share the same main menu to access all parts of the system, to modify objects in the R/3 Repository via the ABAP/4 Development WorkBench, to make changes to customising settings using the IMG, to maintain vendor master records, to record payroll time sheet data and to enter General Ledger Account Postings. While setting up the system landscape to segregate development and production activities may reduce the risk of unauthorised changes to programs and customising settings, access controls must be implemented to prevent such changes in the production environment and to restrict the actions of end-users in conformance with their assigned roles.

Access controls should be designed to restrict the functions that may be performed by a user. R/3’s authorization system allows the assignment of profiles to users. These profiles are collections of authorizations, eg. to create a general ledger master record. Through the use of a security model, based on defined organisational roles and their defined profiles, access to the various system functions can be restricted to appropriate users. Segregation of duties can be implemented effectively through these mechanisms.

3.0 CONTROL RISK ASSESSMENT IN SAP R/3

The study of a client’s internal control structure and the resultant assessment of control risk are important in planning an audit. *AUS 402 Risk Assessments and Internal Controls* requires the auditor to “... obtain an understanding of the internal control structure sufficient to plan the audit and develop an effective audit approach”. This section addresses how to meet this requirement in an SAP R/3 environment.

AUS 402 defines ‘control risk’ as ‘... the risk that misstatement could occur in an account balance or class of transactions that could be material, individually or when aggregated with misstatements in other balances or classes, and that will not be prevented or detected on a timely basis by the internal control structure’. *AUS 214 Auditing in a CIS Environment* states that the auditor should assess both inherent and control risks for material financial report assertions. ‘Inherent risk’ is defined by AUS 402 as “... the susceptibility of an account balance or class of transactions to misstatement that could be material, individually or when aggregated with misstatements in other balances or classes, assuming that an internal control structure did not exist”. The manner in which control risk interacts with audit risk, inherent risk and detection risk is defined in AUS 402.

In assessing inherent risk, the auditor should consider: the organisation of the CIS activities within the organisation and the consequent extent of segregation of duties; the significance of computer processing in each significant accounting application; the complexity of computer processing in each application in terms of volume of transactions, automatic generation of transactions, complicated calculations performed automatically, and exchange of electronic transactions with other organisations; significant changes in IT systems and sufficiency of IT system skills and resources [AUS 402; Wright (1999)]. In many SAP R/3 audit environments, consideration of these factors would lead to inherent risk being assessed as relatively high. Segregating the duties of users is an extremely complex process in R/3 and is prone to error in designing the organisation's security model and implementing profiles. ERP systems are 'enterprise-wide' in scope incorporating potentially all significant accounting applications and involving huge volumes of transactions. The R/3 system facilitates automatic generation of transactions such as adjustments for foreign exchange transactions, postings from sub-ledgers to the general ledger, calculation and posting of goods and services taxes (GST), recording discounts taken by customers, payment of vendor invoices when due, and tracing expenses to cost centres. SAP R/3 also provides for direct debits from customer bank accounts and electronic data interchange (EDI) transactions with vendors. Incorrect configuration for these capabilities may also be subject to error.

In assessing control risk, the auditor must assess both general controls and application controls. **General controls** establish a framework of overall control over information systems activities. They typically include controls over organisation and management of information systems, operations, logical access, physical access, continuity of operations, and software development, acquisition and maintenance. **Application controls** establish specific control procedures over applications to provide reasonable assurance that transactions are authorised and recorded, and are processed completely, accurately and on a timely basis. They are often classified as input controls, controls over data files, controls over processing and output controls. They may be manual controls or incorporated within the IT system.

The strength of general controls can affect the reliability of application controls. Weaknesses in pervasive general controls such as system development and program maintenance, and the control over users' access to sensitive functions, may cause control risk to be assessed at a high level because of the increased risk that application controls may be undermined or not be effective [AUS 402]. Where system development and program maintenance is not controlled adequately, there is the risk of poorly developed, tested and documented programs, and the risk of unauthorised program modifications. Where users' access is not controlled adequately, segregation of duties may not be effective, resulting in unauthorised transaction data entry.

The assessment of control risk in an SAP R/3 environment is discussed in the following sections. The assessment of application controls and selected pervasive general controls will be discussed.

3.1 ASSESSING APPLICATION CONTROLS

Australian auditing standards do not provide a framework for assessing application controls. Several frameworks are proposed in the auditing literature and used by practitioners. Arens et. al. (1997) proposes a framework to be applied at the individual transaction type level:

1. Recorded transactions are valid (validity).
2. Transactions are properly authorised (authorization).
3. Existing transactions are recorded (completeness).
4. Transactions are properly valued (valuation).
5. Transactions are properly classified (classification).
6. Transactions are recorded at the proper time (timing).
7. Transactions are properly included in master files and correctly summarised (posting and summarisation).

Utilising this internal control framework for a specific type of transaction, the auditor can document the internal controls implemented by a client. Exposures (absence of appropriate controls) can be identified where internal control objectives are not satisfied adequately. The auditor can then plan tests of controls and substantive tests of transactions based on this framework. Substantive procedures designed to test audit assertions (objectives) for related account balances can also be planned.

This framework for assessing application controls can be applied to the audit of SAP R/3 transactions. This will be demonstrated with reference to general ledger postings. In R/3, three categories of application controls may be distinguished: (1) **manual** application controls, such as the review of documents before data entry and the review of output from the system; (2) **SAP automatic** controls – these are mandatory controls executed automatically by the system; and (3) **SAP configurable** controls – these are discretionary customising settings, such as specifying the general ledger account for posting GST collected from customers.

Tables 1 to 5 demonstrate the application of this internal control framework to general ledger postings. Table 1 provides examples of internal controls relevant to the validity, valuation and classification objectives. Even in ERP systems, auditors may place reliance on the controls executed manually by users. In general ledger postings, the independent review on a daily basis of journals summarising the general ledger posting documents represents a strong control. The automatic controls provided by SAP R/3 are extensive. Postings are restricted to general ledger accounts by the document type and posting keys. Input fields are checked to verify that all required data is entered and validity checks are satisfied. Account numbers must exist in the Chart of Accounts. Postings of transactions to sub-ledgers (Accounts Receivable and Accounts Payable) result in automatic postings to the corresponding general ledger reconciliation (control) accounts. Direct postings to these reconciliation accounts are prohibited.

Users can review their debit and credit line items before posting to check that they balance to zero. Up to 999 line items can be included in a single general ledger posting document. A

document will be posted only if total debits balance with total credits. Automatic postings are also performed by the system, taking care of the calculation and posting of GST on inputs and outputs. Gains or losses on exchange rate differences involving monetary items are posted automatically to appropriate general ledger accounts. While R/3 permits the parking or holding of incomplete documents, account balances are updated only on the release and balancing of these documents. Changes to posted documents are permitted but are severely limited, thereby preserving the integrity of the transactions. Where documents have been posted containing incorrect data, a document reversal facility 'backs-out' the transaction, avoiding the need to post adjustments.

SAP configurable controls are customising settings that prescribe how the system should operate to meet the organisation's needs. To handle the calculation and reporting of GST on transactions, the organisation must determine what GST tax codes must be defined and the general ledger accounts (receivable and payable) to which GST amounts are to be posted. Where a general ledger account is not 'tax-relevant', the field Posting Without Tax Permitted may be set. The Tax Category field for revenue and expense accounts may be set as relevant to Output Tax and Input Tax respectively, to reduce the incorrect selection of GST tax codes during data entry. Field status definitions may be determined which prescribe whether input fields are to be required, optional or suppressed. With the introduction of the GST, these definitions may need to be modified to ensure that the tax code becomes a required field during transaction entry.

Validations are organisation-specific, additional checks on input data applied to the document header or line items. A user-defined Warning or Error message is specified. Validations may consist of up to 999 steps. Each step has a prerequisite statement (determining whether an entered value should be checked) and a check statement (determining whether an entered value is valid). If the prerequisite is not true, then the data is valid. If the prerequisite is true, then the data is to be checked. If the check is true, then the data is valid. If the check is false, the message will be displayed. Substitution is the process of checking entered values against user-defined statements. If the statement is true, the system substitutes new values for the entered values. A substitution can consist of up to 999 steps. Each step has a prerequisite statement (determining whether an entered value should be substituted), and a substitution value (the value with which the incoming value should be substituted). For example, if the cost centre is '10', then substitute the value '1' for the plant and value '3' for business area. The entered values are replaced by these values. Substitutions are performed before validations.

Where transactions are entered involving foreign currencies, R/3 refers to its table of exchange rates to convert amounts into local currency (the amounts are also stored in the foreign currency). Alternatively, the current exchange rate can be entered with the transaction. To detect data entry errors, a maximum exchange rate difference (tolerable percentage) can be set which will alert the user to intolerable differences between the rate entered and that stored in the exchange rate table. R/3 also accommodates the organisational requirement for recurring documents ('standing journal entries'). Routine transactions (eg. rent payments) which are the same in each posting period may be defined and scheduled for posting.

The auditor tests manual controls in traditional ways, by examining source documents and reports for evidence that the control has been performed (eg. Reviewing signatures). SAP automatic controls are built into the programs and may be tested whenever changes are made. Auditors may review the adequacy of test data and test results, and perform further tests to obtain reasonable assurance that these controls are effective. R/3 provides a computer aided testing tool (CATT) to assist in the application of test data to programs. This tool facilitates the creation of a set of test transactions that may be applied to programs each time they are modified. Consistency in test results between different program versions may be evaluated. SAP configurable controls may be reviewed by examining the customising settings with the IMG. The operation of these controls can also be tested using test data and CATT. Generally, it is more practical for internal auditors to perform many of these tests. External auditors may help plan the internal audit program for a client and may modify the nature and extent of their audit procedures based on the work of internal auditors (AUS 604).

3.2 ASSESSING PERVASIVE GENERAL CONTROLS

This section examines the assessment of control risk in two areas – system development and program maintenance, and controlling user access to sensitive functions. Both of these general controls are considered pervasive, since weaknesses in these controls can undermine the operation of application controls or make them ineffective.

3.2.1 Control over System Development and Program Maintenance

Poor controls over system development and program maintenance may result in improperly developed and tested program changes, undetected unauthorised program changes, and changes made to production programs. These risks may be reduced through the implementation of formal change procedures, adhering to the following principles [Watne & Turney (1990); Weber (1999)]:

- Program changes should be approved by users, internal audit and information systems management.
- Program change requests should be authorised.
- Program change requests should be documented fully.
- Development of program changes should be permitted only for approved and authorised change requests.
- Program changes should comply with programming standards.
- Changes should be restricted to authorised personnel.
- Design specifications should be reviewed and approved by users and internal audit.
- Changes should be made in a development environment not in production.
- All changes should be tested thoroughly before implementation.
- Changes and test results should be reviewed and approved independently.
- Users should be retrained as required.

- All systems, operations and user documentation should be updated.
- Control should be exercised over the transfer of the changed program to production.

In auditing controls over system development and program maintenance in an SAP R/3 environment, the following issues should be addressed [Institute of Internal Auditors (1997), Australian National Audit Office (1998)]:

- Are separate systems or instances maintained for development, quality assurance and production?
- Are changes to programs and customising settings restricted to the development system? Where test results prove unsatisfactory, further changes must be made in the development system and then re-transported to quality assurance for further testing.
- Does the CTS control the transport of all changes from development to quality assurance, and to production?

Both application programs and business data are stored in the R/3 database, along with documentation. This database has two logical components – the R/3 Repository and customer data. The R/3 Repository stores the data structures (the data dictionary) and programs needed to maintain the data in the system. Customer data includes customising settings, application data (eg. general ledger data) and user master data (user passwords and profiles). Within the R/3 Repository, subdivisions can be set up called **clients**. Users must specify a client during login. Each client has its own data, but accesses the same R/3 Repository objects. Most customer data is client-dependent. Some customising settings may be client-independent.

SAP recommends a minimum of three clients [Metzger & Roehrs (2000)]. **CUST** is the customising and development client. All changes performed in this client are documented and recorded in change requests, so they can be promoted to other clients in the system landscape. **QTST** is the quality assurance client for testing customising settings and application functionality. **PROD** is the production client, where the organisation records its transactions. Customising and development performed in CUST should be promoted to PROD only after careful testing in QTST.

SAP also recommends that unit testing of a particular program, be performed in a **UTST** (unit testing) client rather than in CUST, thereby keeping CUST free of application data needed for this testing. A range of unit testing clients may be used to test reports, screens, etc. Other useful clients are a **SAND** (sandbox) client where new customising settings are trialed and a **TRNG** (training) client.

Clients are assigned:

1. A client ID number, eg. 400.
2. A short description.
3. Default currency.
4. The client role, eg. Production, Test, Customising.
5. Client-dependent change option.

6. Client-independent change option.
7. Client protection and restrictions, eg. protection against overwrites and upgrades.

The most critical client setting is the **client-dependent change option** which determines whether changes are permitted in a client and whether changes are going to be recorded automatically to change requests. These settings are:

1. **Changes without automatic recording.** This allows changes to client-dependent customising, but does not automatically record the changes in change requests. Customising changes can be manually included in change requests for promotion to other clients. This approach is not recommended because of the difficulty of keeping track of such changes.
2. **Automatic recording of changes.** This allows changes to client-dependent customising and automatically includes them in change requests. This setting is assigned to CUST.
3. **No changes allowed.** This prevents changes to client-dependent customising. This setting is assigned to UTST, QTST, TRNG and PROD.
4. **No transports allowed.** This allows client-dependent customising changes that will not be promoted to other clients. This setting is assigned to SAND.

The change options for **client-independent** attributes protect both client-independent customising and R/3 Repository Objects. These settings are:

1. **Changes to Repository and client-independent customising allowed.**
2. **No changes to client-independent customising objects (changes to Repository Objects are still allowed).**
3. **No changes to Repository objects (changes to client-independent customising are still allowed).**
4. **No changes to Repository and client-independent customising objects.**

All clients except CUST should be assigned **No changes to Repository and client-independent customising objects**. Such changes are permitted in CUST. SAND should not be used to test client-independent customising changes, only client-dependent changes.

Because of the functional and technical limitations of multiple clients in a single R/3 system, SAP recommends the distribution of these critical R/3 clients among at least three systems, as illustrated in Table 6. Client-dependent customising changes can be tested in SAND before being implemented in CUST. UTST must be in the same system as CUST. Once changes have been unit tested, they can be promoted to QTST. Having TRNG in QAS ensures that training activities do not impact on performance in the production system. More complex landscapes are feasible, with multiple unit testing clients, quality assurance clients and training clients [Metzger & Roehrs (2000)].

Change requests are used to update other clients within the system landscape. This involves the technical procedure for transport, consisting of 2 steps: promotion and

import. **Promoting** changes involves releasing changes and exporting them out of their R/3 system to the operating system level. They are then **imported** into another system. The key to effective transport of changes is maintenance of an orderly system landscape. **Transport routes** must first be defined using the Transport Management System (TMS). Transport routes are necessary from DEV to QAS, and from QAS to PRD, and between clients, eg. from CUST to SAND, from CUST to QTST.

These controls should be tested by the auditor by examining table settings, and following through samples of program changes to ensure appropriate authorization, testing, and transport.

3.2.2 Control over User Access

A fundamental internal control for minimising loss through fraud and error is the proper segregation of duties among personnel [Albrecht, Howe & Romney (1984)]. Research into internal controls and external auditors' judgments has indicated that the separation of duties is a dominating factor in the evaluation of an internal control system [Ashton (1974); Ashton and Brown (1980); Hamilton and Wright (1982)]. Segregation of duties in an SAP R/3 environment is accomplished through careful assignment of authorizations and profiles to users. This section provides an overview of how authorizations are administered and audited in an R/3 system.

3.2.2.1 The Authorization Concept

R/3 uses authorization objects to assign authorizations to users. An authorization object is a template for an authorization. For example, authorization object F_SKA1_BUK - *G/L Account: Authorization for company codes* requires the specification of two field values: Company Code and Activity. To allow a General Ledger supervisor to create a general ledger master record, he/she must be assigned an authorization to create (Activity 1) accounts for a specific company code (eg. Company Code 2000). Such an authorization is created using the object F_SKA1_BUK by assigning these field values and naming the authorization following an appropriate convention (eg. Z_SCC20001).

Authorizations may be classified as general authorizations, organisational authorizations or functional authorizations. General authorizations specify the functions a user may perform. Authorization object F_SKA1_BUK has been assigned to the function for creating general ledger master records. The system checks for the user's authorization to create general ledger accounts (Activity 1) in at least one company code. The system then checks whether the user is permitted to create accounts for the specified organisational unit (company code) and has the required functional authorizations. Authorizations in this case may restrict the user to certain Charts of Accounts. In addition, an authorization group may be defined in certain authorization objects to protect individual master records.

Profiles relating to an organisational role (eg. General Ledger Supervisor) are defined consisting of a list of authorizations and other profiles. Such profiles are then assigned to users with that role and stored in their user master record along with other data (eg. password).

3.2.2.2 Implementing Authorizations

Proper implementation of authorizations is a critical ingredient for the maintenance of security in an R/3 system. Accordingly, it demands an appropriate formalised process. A preferred approach involves the following steps:

1. Defining organisational roles.
2. Identification of the R/3 functions (menu options) associated with each role.
3. Identification of the authorizations required for each function.
4. Designing authorizations and profiles.
5. Creating authorizations and profiles in the development system.
6. Testing authorizations and profiles in the quality assurance system.
7. Transporting authorizations and profiles to the production system.
8. Assigning profiles to user master records.

Steps 1 and 2 are best accomplished using a security model documented in the form of a table. The R/3 functions (menu options) associated with each role should also show the Transaction Code assigned to each function. For example, the Transaction Code for creating a general ledger master record is FS01. Transaction codes are 'short-cuts' to menu options and are the link to the authorizations required for each function. R/3 table USOBT contains a list of authorization objects and field values required for each Transaction Code. User-defined (additional to SAP-defined) objects are documented in table TOBJ.

Steps 3 to 5 may be performed as summarised above. Alternatively, from release 3.1G authorizations may be implemented using the Profile Generator. This software uses a graphical interface to identify and assign field values to authorizations. Activity groups (similar to profiles) are defined and permitted menu options (Transaction codes) are selected. Field values for authorizations are proposed by the software or can be entered. Activity groups are assigned to users as with profiles.

Predefined activity groups (PdAGs) are available through the R/3 Simplification Group to reduce security implementation time, cost and complexity. PdAGs have relevant authorization data already defined and are provided for standard basis administration roles, such as developer (programmer) and customiser, and for end-user roles, such as financial accounting clerk. These activity groups are imported into the system and assigned to users. PdAGs may provide savings but may not meet the organisation's requirements for satisfactory segregation of duties.

3.2.2.3 Auditing Authorizations

Determining whether authorizations have been properly assigned to users may be quite a challenge to auditors. There are over 600 Transaction Codes in the R/3 FI – Financial Accounting Module alone, each requiring its own set of authorizations. Authorizations may have been implemented using the system’s user maintenance transactions, the profile generator, predefined activity groups or some combination. Authorizations and profiles created using user maintenance cannot be maintained using the profile generator.

Auditors typically plan to evaluate and compliance test the client’s security model [Institute of Internal Auditors (1997); Ernst & Young (1995)]. This approach is appropriate when the client has implemented authorizations in a structured, well-documented manner. The security model is ‘desk-checked’ for completeness and proper segregation of duties, and then tested for proper implementation on a ‘sample’ basis by interrogating authorizations, profiles (or activity groups) and user master records. All users with the same role should be assigned the same authorizations and profiles (activity groups). Proper segregation of organisational responsibilities is a critical concern in this process.

Another approach to auditing authorizations involves the interrogation of system tables (including tables USOBT, USR04, USR10 and USR12) to identify authorizations assigned to users and the corresponding Transaction Codes which may be executed. A software application may be developed (eg. using MS ACCESS) to produce a Transaction Code ‘profile’ for each user. Profiles for users with the same role can be compared for discrepancies. Authorizations for incompatible functions may then be revealed.

Standard R/3 reports provided by the Authorizations Information System may also prove useful for auditing authorizations. For example, RSUSR005 lists users with ‘critical’ Basis authorizations; RSUSR008 lists users with ‘critical’ combinations of authorizations (Transaction Codes); RSUSR010 lists Transaction Codes that may be executed by a specified user, profile, authorization or authorization object; and RSUSR050 compares two user master records, profiles or authorizations. Changes to users, profiles and authorizations may also be reviewed using standard reports. The functions that may be performed by a user with a specified role may be listed and investigated. This user’s master record may then be compared with other users with the same role to highlight differences.

3.0 RESEARCH OPPORTUNITIES

The performance of an audit in a sophisticated and technically complex environment such as SAP R/3 presents significant challenges. Internal control and auditing objectives

remain the same in this environment. However, grappling with the scope and technical detail places heavy demands on the auditor. To illustrate, contractors who assist in the implementation of R/3 rarely themselves understand fully the workings of all parts of the system. Consultants usually specialise in one or two modules, such as Financial Accounting and Controlling, or Sales & Distribution and Materials Management. Somehow, the audit team in combination needs to acquire the same level of knowledge as these consultants in order to communicate with client staff, identify and document internal controls, test those controls and perform other audit procedures.

Research opportunities in several areas relevant to audit practice can be identified. The importance of segregation of duties has been discussed. Several basic principles for segregating incompatible functions have been proposed by various writers [Watne & Turney (1990); Srinidhi (1994); Arens, et. al. (1997); and Institute of Internal Auditors (1997)], including:

1. The roles of systems personnel should be segregated from those of end-users;
2. No changes to programs and customising settings should be permitted in the production environment;
3. There should be segregation of duties among systems personnel, eg. system administrator, developer, customiser, authorization administrator, user administrator, etc.;
4. Functional areas should be segregated, eg. accounts receivable, accounts payable, payroll;
5. Those with the custody of assets should not account for them;
6. Those who authorise transactions should not have custody of related assets;
7. Those who authorise transactions should not execute them;
8. Areas of operational activity should be segregated from accounting;
9. Master file maintenance and transaction data entry should be segregated.

Segregation of duties within an R/3 system can be tightly controlled as described above. An interesting research question, however, is how the thousands of functions (Transaction Codes) should be segregated. As a minimum, should there be at least three distinct roles in Accounts Payable – accounts payable supervisor (controlling payments), vendor manager, and invoice data entry? Who should perform period-end processing – the accountant, general ledger supervisor, or someone else? In the FI - Financial Accounting Module, there are over 600 such functions to be assigned to personnel. Anecdotal and preliminary case study research evidence indicates that there is little consensus among practitioners on how these functions should be segregated.

There has been a great deal of research interest in the development of decision support tools for auditing, computer security and related areas [Connell (1987); Abdolmohammadi (1987); Denning (1987); Smaha (1988); Quinlan (1989); Hellerstein, Klein & Milliken (1990); McAuliffe et. al. (1990); van Dijk & Williams (1990); Jagannathan et. al. (1993); Best, Mohay & Anderson (1997); Mounji & Le Charlier (1997); Daniels & Spafford (1999); and MIT's DARPA Intrusion Detection Evaluation Project (<http://www.ll.mit.edu/IST/ideval/index.html>)]. The feasibility of auditing authorizations

through the interrogation of system tables was discussed above. Such a software application may be used to determine what authorizations have been assigned to users and the related functions that they can perform. The development of such a decision support system for auditors would represent a valuable research project and make a practical contribution to the auditing profession, particularly if able to test for proper segregation of duties as discussed above.

Further research opportunities focus on R/3 audit trails. The R/3 system maintains an extensive array of audit trails incorporating a combination of operating system and accounting audit trail characteristics. Changes to authorizations, profiles and user master records may be examined and checked for propriety using standard reports. Changes to fields in master records may be examined. The documents underlying account balances may be reviewed by 'drilling down' from account balances, to line items (individual postings), and to the original documents. Document headers show document date and posting date, as well as identification of the user who entered the data. This data may also be included in reports created using R/3 tools such as ABAP Query, Report Writer and Report Painter.

In the context of auditing authorizations, user monitoring is feasible using standard R/3 performance monitoring reports. Analyses of this audit trail over a period of time would permit the review of the functions (Transaction Codes) actually performed by users in contrast to those for which they should have authorizations. These actual user 'profiles' can be compared with the client's security model which defines the roles and associated functions to be assigned to users. Such comparisons may highlight deficiencies in the implementation of authorizations and profiles. It is acknowledged however that apparent consistency between users' actions and their intended profiles according to the security model does not guarantee the correct implementation of authorizations and profiles.

A knowledge base system may be developed to generate actual user 'profiles' and forecasts of expected user activity. Such profiles characterise the behaviour of a given user with respect to various objects, representing a definition of normal activity and a means of detecting anomalous activity. Changes in actual user behaviour may be detected promptly and investigated. Anomalous user behaviour may be associated with unauthorised activity, including browsing the system or masquerading as another user [Smaha (1988)]. Such a knowledge base system is currently under development by extending the MIATA system which tested the feasibility of machine-independent audit trail analysis in mainframe environments [Best, Mohay & Anderson (1997)]. This project examines the research proposition that R/3 user behaviour is forecastable (on the basis of minimising forecast error and providing a 'good fit') and may be categorised as normal or anomalous.

4.0 CONCLUSION

This paper has provided an introduction to the audit of the SAP R/3 system and some appreciation of the challenges facing the auditor. The assessment of control risk was

discussed focusing on the application of an internal control framework in the assessment of application controls and the assessment of two significant pervasive general control areas. Several opportunities for research were outlined.

BIBLIOGRAPHY

Abdolmohammadi, M.J. (1987) "Decision Support and Expert Systems in Auditing: A Review and Research Directions", *Accounting and Business Research*, Vol. 17, No. 66, pp. 173-185.

Albrecht, W., Howe, K. & Romney, M. (1984) *Deterring Fraud: The Internal Auditor's Perspective*, The Institute of Internal Auditors Research Foundation, Altamonte Springs, Florida.

Arens, A.A., Best, P.J., Shailer, G.E.P. & Loebbecke, J.K. (1997) *Auditing in Australia: an Integrated Approach*, Fourth Edition, Prentice-Hall, Sydney.

ASAP World Consultancy (1997) *Using SAP R/3*, Second Edition, Que, Henley on Thames.

Ashton, R. (1974) "An experimental study of internal control judgements", *Journal of Accounting Research*, Vol. 12, No. 1, Spring, pp. 143-157.

Ashton, R. & Brown, R. (1980) "Descriptive modelling of auditor's internal control judgements: replication and extension", *Journal of Accounting Research*, Vol. 18, No. 1, Spring, pp. 269-277.

Australian National Audit Office (1998) *Security and Control for SAP R/3*, Commonwealth of Australia.

Best, P.J., Mohay, G. & Anderson, A. (1997) "MIATA : A Machine Independent Audit Trail Analyser", *Australian Computer Journal*, Vol. 29, No. 2, May, pp. 57-63.

Connell, N.A.D. (1987) "Expert Systems in Accountancy: A Review of Some Recent Applications", *Accounting and Business Research*, Vol. 17, No. 67, pp. 221-233.

Daniels, T.E. & Spafford, E.H. (1999) "Identification of Host Audit Data to Detect Attacks on Low-Level IP Vulnerabilities", *Journal of Computer Security*, Vol. 7, No. 1, pp. 3-35.

Denning, D.E. (1987) "An Intrusion Detection Model", *IEEE Transactions on Software Engineering*, Vol. SE-13, No.2, pp. 222-232.

Ernst & Young (1995) *Audit, Control, and Security Features of SAP R/3*.

Gilbert , A. (2000) "ERP Vendors Look at Rebound after Slowdown", *Informationweek*, No. 773, February 14, pp. 98-100.

Hamilton, R. and W. Wright. (1982) "Internal control judgments and effects of experience: replications and extensions", *Journal of Accounting Research*, Vol. 20, No. 2, Autumn, pp. 756-765.

Hellerstein, J.L., Klein, D.A. & Milliken, K.R. (1990) *Expert Systems in Data Processing*, Addison-Wesley, Reading, MA.

Institute of Internal Auditors (1997) *SAP R/3: Its Use, Control, and Audit*, Institute of Internal Auditors Research Foundation, Altamonte Springs, Florida.

Jagannathan, R., Lunt, T., Anderson, D., Dodd, C., Gilham, F., Jalali, C., Javitz, H., Neumann, P., Tamaru, A. & Valdes, A. (1993) *System Design Document: Next-Generation Intrusion Detection Expert System (NIDES)*, Technical Report, SRI International, Menlo Park, CA.

McAuliffe, N., Wolcott, D., Schaefer, L., Kelem, N., Hubbard, B. & Haley, T. (1990) "Is Your Computer being Misused? A Survey of Current Intrusion Detection System Technology", *Proc. 6th Annual Computer Security Applications Conference*, IEEE Computer Society Press, December, Tucson, Arizona, pp. 260-272.

Metzger, S.M. & Roehrs, S. (2000) *SAP R/3 Change and Transport Management: The Official SAP Guide*, Sybex, London.

Mounji, A. & Le Charlier, B. (1997) "Continuous Assessment of a Unix Configuration: Integrating Intrusion Detection and Configuration Analysis", in *Proceedings of the ISOC'97 Symposium on Network and Distributed System Security*, San Diego, California.

Norris, G., Wright, I., Hurley, J.R., Dunleavy, J. & Gibson, A. (1998) *SAP: An Executive's Comprehensive Guide*, Wiley, New York.

Quinlan, J.R., ed. (1989) *Applications of Expert Systems: Volume 2*, Addison-Wesley, Sydney.

Smaha, S.E. (1988) "Haystack: An Intrusion Detection System", *Fourth Aerospace Computer Security Applications Conference*, pp. 37-44.

Srinidhi, B. (1994) "The influence of segregation of duties on internal control judgements", *Journal of Accounting, Auditing & Finance*, Vol. 9, No. 3, pp. 423-444.

van Dijk, J.C. & Williams, P. (1990) *Expert Systems in Auditing*, Stockton Press, New York.

Watney, D.A. & Turney, P.B.B. (1990) *Auditing EDP Systems*, Second Edition, Prentice-Hall, Englewood Cliffs, NJ.

Weber, R. (1999) *Information Systems Control and Audit*, Prentice-Hall, Upper Saddle River, NJ.

Wright, M. (1999) *Audit Guide No. 5 - Auditing in an IT Systems Environment*, Australian Accounting Research Foundation, Caulfield, Vic.

Table 1 – Controls for General Ledger Postings - Validity, Valuation and Classification

<i>Category</i>	<i>Examples of Relevant Controls</i>
Manual controls	Data entered from a source document, independently reviewed prior to data entry. Independent review of daily general ledger posting reports (compact journal RFBELJ00 or document journal RFBELJ10).
SAP automatic controls	Default document type – SA (general ledger posting), determines accounts that may be posted and document numbering. Posting keys – 40 (debit); 50 (credit), determine accounts that may be posted. Default data – eg. date. Formal field checks – mandatory fields, correct date format. Field-specific checks – correct type, length, valid value (reference to a check table), valid field combination (document type, posting key and account type). Chart of accounts. Automatic posting of sub-ledger transactions to reconciliation accounts. No posting to reconciliation accounts permitted. Document overview – display of line items permits user review prior to posting. Balance verification – document must balance to zero. Posting period and fiscal year must be open. Automatic postings – for GST, gain/loss on exchange rate differences, cross-company code transactions. Parked and held documents do not update balances. Restricted changes to posted documents – may not change posting key, account number, amount. Document reversal permitted.
SAP configurable controls	Definition of tax codes for GST. Definition of general ledger accounts for posting GST. Posting without tax permitted. Field status definitions – required, optional, suppressed. Validations. Substitutions. Maximum exchange rate differences. Recurring documents.

Table 2 – Controls for General Ledger Postings - Authorization

<i>Category</i>	<i>Examples of Relevant Controls</i>
Manual controls	Data entered from a source document, independently reviewed prior to data entry. Use of park and post strategy. Independent review of daily general ledger posting reports (compact journal RFBELJ00 or document journal RFBELJ10). Review of recurring documents using report RFDAUB00.
SAP automatic controls	The system checks the user's general, organisational and functional authorizations to execute the transaction.
SAP configurable controls	Authorizations assigned to users: <ul style="list-style-type: none"> • to perform the given function, for company code, for document type, for account type • segregating parking and posting of documents, segregating master data maintenance and general ledger posting • restricting access to transaction FB01 (allows posting to any accounts – general ledger, customer, etc.) • restricting access to change currency table • restricting access to change customising settings. Setting tolerance limits in terms of maximum posting amounts. Setting payment difference limits for handling underpayment and overpayment of accounts. Blocking posting to accounts marked for deletion.

Table 3 – Controls for General Ledger Postings – Completeness

<i>Category</i>	<i>Examples of Relevant Controls</i>
Manual controls	Prenumbering of source documents. Recording SAP-generated document number on source document. Proper filing of source documents. Accounting for prenumbered source documents. Review of parked and held documents using RFVBER00. Review of missing document numbers using report RFBNUM00.
SAP automatic controls	Formal field check – mandatory fields.
SAP configurable	Field status definitions – required, optional, suppressed.

controls	Recurring documents. Control totals, displayed/checked with FB07.
----------	--

Table 4 – Controls for General Ledger Postings – Timeliness

<i>Category</i>	<i>Examples of Relevant Controls</i>
Manual controls	Procedures to ensure transactions recorded on a timely basis. Review of parked and held documents. Review of missing document numbers using report RFBNUM00. Review of long outstanding and unusual items using report RFOOPO000.
SAP automatic controls	Warning messages – if posting to an earlier fiscal year. Posting period and fiscal year must be open.
SAP configurable controls	Recurring documents Closing posting period after a predefined number of days. Opening posting periods for posting. Proper carry-forward of year-end balances.

Table 5 – Controls for General Ledger Postings – Posting & Summarisation

<i>Category</i>	<i>Examples of Relevant Controls</i>
Manual controls	Procedures governing day-, month- and period-end processing. Review of reports for complete and accurate posting, including: <ul style="list-style-type: none"> • GL accounts list (RFSKVZ00) • GL line items for open-item accounts (RFSOPO00) • GL account balances (RFSSLD00) • GL balance sheet/P&L (RFBILA00) • Compact journal (RFBELJ00) • Document journal (RFBELJ10) • Posting totals (RFBUSU00).
SAP automatic controls	Automatic posting of sub-ledger transactions to reconciliation accounts. No direct posting to reconciliation accounts. Document overview – permits user review prior to posting. Balance verification – document must balance to zero. Posting period and fiscal year must be open. Automatic postings – for tax, gain/loss on exchange rate differences, cross-company code transactions.
SAP configurable	Closing posting period after a predefined number of days.

controls	Opening posting periods for posting. Proper carry-forward of year-end balances.
----------	--

Table 6 Distribution of Clients among Systems

<i>System</i>	<i>Client(s)</i>	<i>Comments</i>
DEV	CUST UTST SAND	Development system
QAS	QTST TRNG	Quality assurance system
PRD	PROD	Production system