

UNIVERSITY OF SOUTHERN QUEENSLAND

**A CONCEPTUAL MODEL FOR PROACTIVE DETECTION
OF POTENTIAL FRAUD IN ENTERPRISE SYSTEMS:
EXPLOITING SAP AUDIT TRAILS TO DETECT
ASSET MISAPPROPRIATION**

A dissertation submitted by
KISHORE HARICHUNDER SINGH

For the award of
Doctor of Philosophy

2012

ABSTRACT

Fraud costs the Australian economy approximately \$3 billion annually, and its frequency and financial impact continues to grow. Many organisations are poorly prepared to prevent and detect fraud. Fraud prevention is not perfect therefore fraud detection is crucial. Fraud detection strategies are intended to quickly and efficiently identify frauds that circumvent preventative measures so that an organisation can take appropriate corrective action.

Enhancing the ability of organisations to detect potential fraud may have a positive impact on the economy. An effective model that facilitates proactive detection of potential fraud may potentially save costs and reduce the propensity of future fraud by early detection of suspicious user activities. Enterprise systems generate millions of transactions annually. While most of these are legal and routine transactions, a small number may be fraudulent. The enormous number of transactions makes it difficult to find these few instances among legitimate transactions. Without the availability of proactive fraud detection tools, investigating suspicious activities becomes overwhelming.

This study explores and develops innovative methods for proactive detection of potential fraud in enterprise systems. The intention is to build a model for detection of potential fraud based on analysis of patterns or signatures building on theories and concepts of continuous fraud detection. This objective is addressed by answering the main question; *can a generalised model for proactive detection of potential fraud in enterprise systems be developed?* The study proposes a methodology for proactive detection of potential fraud that exploits audit trails in enterprise systems. The concept of proactive detection of potential fraud is demonstrated by developing a prototype. The prototype is a near real-time web based application that uses SAS for its analytics processes. The aim of the prototype is to confirm the feasibility of implementing proactive detection of potential fraud in practice. Verification of the prototype is achieved by performing a series of tests involving simulated activity, followed by a full scale case study with a large international manufacturing company. Validation is achieved by obtaining independent reviews from the case study senior staff, auditing practitioners and a panel of experts. Timing experiments confirm that the prototype is able to handle real data volumes from a real organisation without difficulty thereby providing evidence in support of enhancement of auditor productivity. This study makes a number of contributions to both the literature and auditing practice.

CERTIFICATION OF DISSERTATION

I certify that the ideas, experimental work, results, analyses, software and conclusions reported in this dissertation are entirely my own effort, except where otherwise acknowledged. I also certify that the work is original and has not been previously submitted for any other award, except where otherwise acknowledged.

Signature of Candidate

Date

ENDORSEMENT

Signature of Supervisor

Date

Signature of Supervisor

Date

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the guidance, help, support and encouragement from my supervisors, independent reviewers, expert panel members, family and friends. I would like to express my sincere gratitude to my supervisors, Professor Peter Best and Associate Professor Joseph Mula for their professional guidance, assistance and encouragement during this journey. Professor Peter Best provided exceptional technical insight required for this research. Associate Professor Joseph Mula provided outstanding guidance in developing the research and he spent several hours reading and providing exceptional feedback that helped improve the writing and the structure of this dissertation.

I would like to express my thanks to Mr Nalinde Jayasekara and Mr Kamal Manatunga for agreeing to trial the prototype software and for their independent reviews of the prototype. I would also like to thank Mr John J Halliday, Executive Director Advisory, BDO Australia for providing an independent review of the prototype software and for hosting of an expert panel session.

This research would not have been complete without feedback from members of the expert panel. I would like to thank members of CPA Australia (Queensland Division - IT Discussion Group) and ISACA (Queensland Chapter) that participated in the panel, and for their valuable comments and feedback.

My thanks and love is also dedicated to my wife Bharati and my son Akhil. Their encouragement and support gave me strength and persistence to complete this long PhD journey. I would also like to thank all my friends for their support and encouragement.

Finally, I would like to express my humble gratitude to my spiritual master, Sri Sathya Sai Baba, without whose inspiration and guidance, this journey would not have been possible.

TABLE OF CONTENTS

ABSTRACT	i
CERTIFICATION OF DISSERTATION.....	ii
ACKNOWLEDGEMENTS	iii
LIST OF FIGURES	viii
LIST OF TABLES.....	xiii
CHAPTER 1 Introduction.....	16
1.0. Background.....	16
1.1. Research problem.....	19
1.2. Study design	22
1.3. Key definitions.....	23
1.4. Delimitations of scope.....	30
1.5. Research significance	33
1.6. Structure of dissertation.....	35
1.7. Conclusion.....	36
CHAPTER 2 Literature Review	38
2.0. Introduction	38
2.1. Definition of fraud	40
2.1.1. Asset misappropriation.....	42
2.1.2. Corruption.....	46
2.1.3. Fraudulent financial statements	47
2.2. Occurrence and cost of fraud.....	48
2.3. Motivation to commit fraud.....	53
2.4. Framework for perpetrating fraud.....	59
2.5. Fraud scenarios	65
2.6. Proactive fraud detection.....	69
2.7. Continuous monitoring strategies	74
2.7.1. Embedded Audit Modules (EAMs)	77
2.7.2. Monitoring and Control Layer (MCL)	81

2.8.	Enterprise Systems	84
2.9.	SAP Enterprise System	87
2.10.	Audit trails	90
2.11.	Enterprise system audit trails support for fraud detection.....	93
2.12.	Gaps in the literature	97
2.13.	Conclusion	99
CHAPTER 3 Research Design and Methodology.....		102
3.0.	Introduction	102
3.1.	Study design	103
3.2.	Research design	107
3.2.1.	Research questions	107
3.2.2.	Conceptual model.....	109
3.2.3.	Research propositions.....	112
3.3.	Research methodology	118
3.3.1.	Scope of fraud categories	120
3.3.2.	Measures to detect fraud.....	121
3.3.3.	Data requirements to detect potential fraud.....	124
3.3.4.	Prototype.....	127
3.3.5.	Data collection	132
3.3.6.	Proof of conceptual model.....	133
3.3.7.	Expert panel validation of model.....	134
3.4.	Conclusion.....	137
CHAPTER 4 Prototype Design.....		139
4.0.	Introduction	139
4.1.	Prototype design.....	140
4.2.	Data requirements for fraud detection.....	141
4.3.	SAP support for fraud detection	143
4.3.1.	SAP audit trails	144
4.4.	Catalogue of fraud symptoms.....	150
4.4.1.	Critical combinations	151
4.4.2.	Known fraud symptoms	153
4.5.	Design specification for fraud detection strategies.....	154
4.6.	Storage.....	176

4.7.	Output.....	177
4.8.	User interface.....	181
4.9.	Errors.....	183
4.10.	Verification and validation of prototype	184
4.11.	Prototype design and propositions addressed	188
4.12.	Conclusion	190
CHAPTER 5 Prototype Implementation and Testing		192
5.0.	Introduction	192
5.1.	Prototype implementation	193
5.1.1.	Workstation environment	193
5.1.2.	Development environment.....	193
5.1.3.	Data extraction and pre-processing	194
5.1.4.	Reporting system.....	197
5.2.	Verification and testing of prototype	204
5.2.1.	Test data.....	205
5.2.2.	Case study 1a: Data from large international manufacturing company ...	212
5.2.3.	Case study 1b: Subset of case study 1a data.....	218
5.2.4.	Case study 1a: Summary of findings and recommendations.....	220
5.3.	Processing times.....	223
5.4.	Validation and independent review of prototype.....	229
5.5.	Prototype implementation and testing and propositions	236
5.6.	Conclusion.....	238
CHAPTER 6 Conclusion and Further Research		241
6.0.	Introduction	241
6.1.	Summary of results from this study	241
6.2.	Contributions	250
6.2.1.	Theoretical contributions.....	251
6.2.2.	Contributions to the practice of fraud detection	257
6.3.	Limitations.....	259
6.4.	Recommendations.....	261
6.4.1.	Recommendations for further research	261
6.4.2.	Recommendations for extensions to prototype.....	264
6.5.	Conclusion.....	272

LIST OF REFERENCES	274
APPENDICES	287
Appendix 1: Fraud cases	287
Appendix 2: Expert panel protocol	291
Appendix 3: Prototype evaluation questionnaire	293
Appendix 4: Prototype menu navigation	296
Appendix 5: Results from test data	308
Appendix 6: Results from case study 1a	333
Appendix 7: Results from case study 1b	350
Appendix 8: Data extraction	353
Appendix 9: Feedback on prototype	357

LIST OF FIGURES

Figure 2.1: Categories of occupational fraud	43
Figure 2.2: Position of fraud perpetrator	51
Figure 2.3: Fraud cases based on perpetrator's department	51
Figure 2.4: Median loss by perpetrator's department	52
Figure 2.5: Fraud triangle	56
Figure 2.6: Theoretical foundation for research.....	58
Figure 2.7: Fraud perception model (FPM)	61
Figure 2.8: High-level fraud scenarios model (HFSSM).....	63
Figure 2.9: Fraud detection process.....	71
Figure 2.10: Detection of occupational fraud	73
Figure 3.1: Conceptual model	110
Figure 3.2: Research propositions	114
Figure 3.3: Methodology	119
Figure 3.4: Critical AP activities model	122
Figure 3.5: Flipping vendor bank account details	127
Figure 3.6: The prototype model.....	129
Figure 3.7: Prototype input requirements	130
Figure 3.8: Source of data.....	133
Figure 3.9: Expert panel interaction and feedback.....	137
Figure 4.1: Prototype conceptual design.....	141
Figure 4.2: Input specifications	142
Figure 4.3: SAP audit trails	146
Figure 4.4: Process module	151
Figure 4.5: Detection strategy- users violating SoDs principle 1.....	156
Figure 4.6: Detection strategy- users violating SoDs principle 2.....	157
Figure 4.7: Detection – flipping vendor bank account	159
Figure 4.8: Detection – duplicate transactions	160
Figure 4.9: Detection – invoices with round dollar amounts.....	161
Figure 4.10: Detection – invoices below approval limit.....	163
Figure 4.11: Detection – vendor payments exceeding last largest.....	164
Figure 4.12: Detection – use of one time vendors.....	165

Figure 4.13: Detection – vendors with similar names	166
Figure 4.14: Detection – vendors becoming active after long period	167
Figure 4.15: Detection – multiple vendors with different payment details	169
Figure 4.16: Detection – multiple vendors sharing payment details	170
Figure 4.17: Detection – Benford's Law analysis of invoices.....	172
Figure 4.18: Storage module	177
Figure 4.19: Output module	177
Figure 4.20: Visualisation - activity summary	180
Figure 4.21: Visualisation - user profile	180
Figure 4.22: Visualisation – interaction between users and individual vendor	181
Figure 4.23: User-interface	182
Figure 4.24: Verification and validation.....	184
Figure 4.25: Prototype logical design.....	191
Figure 5. 1: User interface.....	195
Figure 5.2: Process map – complete.....	198
Figure 5.3: Analysis process	199
Figure 5.4: Dashboard.....	200
Figure 5.5: User activity reports.....	202
Figure 5.6: Individual vendor reports.....	203
Figure 5.7: Number of records processed	226
Figure 6.1: Model of extended prototype	269
Figure A4.1: Start-up screen	296
Figure A4. 2: Accounts payable summary menu	297
Figure A4. 3: User profiles menu	297
Figure A4. 4: Critical combinations menu.....	298
Figure A4. 5: User activity analysis menu	298
Figure A4. 6: Detailed user activity analysis menu.....	299
Figure A4. 7: User activity reports menu.....	300
Figure A4. 8: Vendor analysis menu	301
Figure A4. 9: Analyse vendor transactions menu	302
Figure A4. 10: Analysis of vendor transactions (reports) menu	303
Figure A4. 11: Search vendor menu.....	304
Figure A4. 12: Configure system menu.....	305

Figure A4. 13: Set date range for analysis menu	305
Figure A4. 14: Set approval limit for invoices & payments menu.....	306
Figure A4. 15: File processing menu.....	306
Figure A4. 16: Data conversion & import menu.....	307
Figure A4. 17: Update/create data warehouse selection screen.....	307
Figure A5.1: Dashboard.....	309
Figure A5. 2: User activities summary	310
Figure A5. 3: User profile – vendor maintenance	310
Figure A5. 4: User profile – invoice transactions	311
Figure A5. 5: User profile – payment transactions.....	311
Figure A5. 6: User profile – invoices or payment transactions.....	312
Figure A5. 7: User profile – all combinations.....	312
Figure A5. 8: Visualisation – all combinations.....	313
Figure A5. 9: Violation of SoDs – users entering invoices and payments	314
Figure A5. 10: Visualisation - users entering invoices and payments.....	314
Figure A5. 11: Violation of SoDs – users performing vendor maintenance.....	314
Figure A5. 12: Visualisation - users performing vendor maintenance.....	315
Figure A5. 13: Violation of SoDs – users performing vendor maintenance.....	315
Figure A5. 14: Visualisation - users performing vendor maintenance.....	316
Figure A5. 15: Violation of SoDs – users performing vendor maintenance,.....	316
Figure A5. 16: Visualisation - users performing vendor maintenance,.....	317
Figure A5. 17: Bank account changes by user - 1USRARSCP	317
Figure A5. 18: Invoice transactions by user - 1USRARSCP.....	318
Figure A5. 19: Payment transactions by user - 1USRARSCP.....	318
Figure A5. 20: Round dollar payments by user - 1USRARSCP.....	319
Figure A5. 21: Vendors touched by user - 1USRARSCP	319
Figure A5. 22: Visualisation – vendors touched by user - 1USRARSCP	320
Figure A5. 23: User 1USRARSCP interacting with vendor 0002000041.....	321
Figure A5. 24: Vendors sharing bank accounts	322
Figure A5. 25: Visualisation - vendors sharing bank accounts.....	323
Figure A5. 26: Vendors with multiple bank accounts	324
Figure A5. 27: Visualisation - vendors having multiple bank accounts.....	325
Figure A5. 28: Vendors with multiple changes.....	326

Figure A5. 29: Vendors with multiple master records	327
Figure A5. 30: Top 5 vendors by sum of invoices	328
Figure A5. 31: Top 5 vendors by sum of payments	328
Figure A5. 32: Benford's Law – analysis of vendor invoices	329
Figure A5. 33: Benford's Law – investigation of spike at digit 49	329
Figure A5. 34: Benford's Law – analysis of vendor payments	330
Figure A5. 35: Benford's Law – investigation of spike at digit 22	330
Figure A5. 36: Transaction history for vendor – showing flipping.....	331
Figure A5. 37: Visualisation - users interacting with vendor	331
Figure A5. 38: Visualisation – vendor transaction history	332
Figure A6.1: Dashboard.....	334
Figure A6. 2: User activities summary	335
Figure A6. 3: User profile – vendor maintenance	335
Figure A6. 4: User profile – invoice transactions	336
Figure A6. 5: User profile – payment transactions.....	336
Figure A6. 6: User profile – invoices or payment transactions.....	337
Figure A6. 7: User profile – all combinations.....	337
Figure A6. 8: Visualisation – all combinations.....	338
Figure A6. 9: Violation of SoDs – users entering invoices and payments	339
Figure A6. 10: Violation of SoDs – users performing vendor maintenance.....	339
Figure A6. 11: Visualisation - users entering invoices and payments.....	340
Figure A6. 12: Visualisation - users performing vendor maintenance.....	340
Figure A6. 13: Violation of SoDs – users performing vendor maintenance.....	341
Figure A6. 14: Violation of SoDs – users performing vendor maintenance,.....	341
Figure A6. 15: Visualisation - users performing vendor maintenance.....	342
Figure A6. 16: Visualisation - users performing vendor maintenance,.....	342
Figure A6. 17: Vendors sharing bank accounts	343
Figure A6. 18: Vendors with multiple bank accounts	343
Figure A6. 19: Visualisation - vendors sharing bank accounts.....	344
Figure A6. 20: Visualisation - vendors having multiple bank accounts.....	345
Figure A6. 21: Vendors with multiple changes.....	346
Figure A6. 22: Vendors with multiple master records	346
Figure A6. 23: Top 5 vendors by sum of invoices	347

Figure A6. 24: Top 5 vendors by sum of payments 347

Figure A6. 25: Benford's Law – analysis of vendor invoices 348

Figure A6. 26: Benford's Law – investigation of spike at digit 36 348

Figure A6. 27: Benford's Law – analysis of vendor payments 349

Figure A6. 28: Benford's Law – investigation of spike at digit 22 349

Figure A7.1: Dashboard..... 350

Figure A7. 2: User activities summary 351

Figure A7. 3: Risky user list 351

Figure A7. 4: Benford's Law – analysis of vendor invoices 352

Figure A7. 5: Benford's Law – analysis of vendor payments 352

LIST OF TABLES

Table 2.1: Categories of occupational fraud and abuse	48
Table 2.2: Sub-categories of asset misappropriation (US Data)	49
Table 2.3: Sub-categories of asset misappropriation (Oceania Data)	50
Table 2.4: Control weakness that contributed to fraud.....	50
Table 2.5: Fraud matrix (FM)	66
Table 3.1: Research paradigms	104
Table 3.1: Mapping of research questions, propositions and process	116
Table 3.2: Methods to detect known fraud symptoms.....	125
Table 4.1: Source of data to detect known fraud symptoms.....	145
Table 4.2: SAP transaction codes.....	152
Table 4.3: Violation of SoDs principle 1	153
Table 4.4: Violation of SoDs principle 2.....	153
Table 4.5: Known AP fraud symptoms	155
Table 4.6: Risk Index variables.....	175
Table 4.7: Activity summary.....	180
Table 5.1: Source of data to detect known fraud symptoms.....	196
Table 5.2: Control values - activities performed by users	206
Table 5.3: Control values - violation of segregation of duties.....	206
Table 5.4: Control values - activities performed by user 1USRARSCP	206
Table 5.5: Control values - summary of vendor transactions	207
Table 5.6: Control values – Benford's Law	207
Table 5.7: Activities performed by users.....	208
Table 5.8: Violation of segregation of duties.....	209
Table 5.9: Activities performed by user 1USRARSCP.....	209
Table 5.10: Summary of vendor transactions	210
Table 5.11: Correspondence between control values and experimental values.....	212
Table 5.12: Activities performed by users.....	213
Table 5.13: Violation of segregation of duties.....	213
Table 5.14: Summary of activities by 1USRA.....	214
Table 5.15: Summary of activities by 1USRMI.....	215

Table 5.16: Summary of activities by 1USREEWAH.....	216
Table 5.17: Summary of activities by 1USRN.....	216
Table 5.18: Summary of vendor transactions	217
Table 5.19: Activities performed by users	219
Table 5.20: Violation of segregation of duties.....	219
Table 5.21: Summary of vendor transactions	220
Table 5.22: Processing time - stage 1 test.....	223
Table 5.23: Number of records processed	224
Table 5.24: Processing time – stage 2 test	224
Table 5.25: Number of records processed	225
Table 5.26: Processing time – stage 3 test	225
Table 5.27: Number of records processed	226
Table 5.28: Summary of records processed for all tests.....	226
Table 5.29: Average processing time for all tests	227
Table 5.30: Operation	231
Table 5.31: Reports.....	232
Table 5.32: Visualisations.....	232
Table 5.33: Accuracy, efficiency and performance.....	233
Table 5.34: Auditor productivity.....	233
Table 5.35: Time to process data manually	234
Table 5.36: Time to process data with other software.....	234
Table 5.37: Time to process data with prototype	235
Table 5.38: Overall evaluation	235
Table A8.1: SAP tables and field requirements	353
Table A8.2: Summary data extraction procedure.....	354
Table A8.3: SAP table extraction documentation	355

CHAPTER 1

Introduction

1.0. Background

According to the Association of Certified Fraud Examiners (ACFE) Report to the Nations on Occupational Fraud & Abuse, "a typical organisation loses five percent of its annual revenue to fraud. Applied to the estimated 2009 Gross World Product of \$58.07 trillion, this figure translates to a potential total fraud loss of more than \$2.9 trillion" (ACFE 2010 p.4). Within Australia this figure is approximately \$3 billion annually (Standards Australia 2008). These figures are clear evidence that fraud is a major problem, which requires serious study by researchers to minimise illegal activities. A fundamental first step in studying the fraud problem is to unambiguously define fraud itself.

There are two principal methods of getting something from others illegally. They can either be physically forced, or they can be deceived into giving up their assets. The first type is called robbery and the second is fraud. Albrecht et al. (2009) defines fraud as a deception made for personal gain. "Deception" is key. The most common definition of fraud according to Webster's Dictionary (2001 p.380) is:

"Fraud is a generic term that embraces all the multifarious means which human ingenuity can devise, which are resorted to by one individual, to get an advantage over another by false representations. No definite and invariable rule can be

laid down as a general proposition in defining fraud, as it includes surprise, trickery, cunning and unfair ways by which another is cheated. The only boundaries defining it are those which limit human knavery."

Australian Government *Fraud Control Guidelines* define fraud as (ComLaw 2011 p.4):

"theft; accounting fraud (false invoices, misappropriation etc); unlawful use of, or obtaining property, equipment, material or services; causing a loss, or avoiding and/or creating a liability; providing false or misleading information, or failing to provide it when there is an obligation to do so; misuse of assets, equipment or facilities; making, or using false, forged or falsified documents; and wrongfully using information or intellectual property."

Furthermore, the Government identifies fraud as targeting revenue, benefits, property, information and intelligence, funding and grants, entitlements, facilities, and money or property. Benefits obtained fraudulently are not restricted to monetary or material benefits, and may be tangible or intangible, including unauthorised provision of access to or disclosure of information. Benefits may also be obtained by third parties in addition to the fraud perpetrator.

Two types of fraud can be distinguished: misappropriation of assets; and fraudulent financial reporting (Casabona and Grego 2003 ; ASB 2002). Misappropriation of assets is often referred to as 'employee fraud' and involves theft of an organisation's

assets. Fraudulent financial reporting involves deliberate misstatements or omissions of amounts or disclosures of financial statements to deceive investors and creditors, increase share price, meet cash flow needs or hide company losses and problems (Romney and Steinbart 2009 ; Wells 2008 ; Casabona and Grego 2003). The ACFE extends this definition by classifying frauds and the methods used to commit them.

The ACFE (2010 p.6) defines occupational fraud as:

"...the use of one's occupation for personal enrichment through the deliberate misuse or misapplication of the employing organization's resources or assets..."

Occupational fraud is very broad and it encompasses a range of transgressions by employees at all levels of an organisational hierarchy. These include i) asset misappropriations, which involve theft or misuse of an organisation's assets; ii) corruption, in which employees wrongfully use their influence in business transactions to gain some benefit for themselves or another person, contrary to their duty to their employer; and iii) fraudulent statements, which usually involve falsification of an organisation's financial statements.

Fraud can be committed by anyone. Perpetrators cannot usually be distinguished from other people on the basis of demographic or psychological factors. Individuals involved in fraud are regular people that have compromised their integrity and become involved in fraud. Several theories exist in the literature as to why individuals commit frauds. A common theme in each of the theories is one of conflict of interest. If this situation arises between the owner(s) and employees, it may lead to

dissatisfaction among employees. Affected employees may seek relief by resorting to fraudulent behaviour when an opportunity presents itself.

Owners incur costs in order to monitor opportunistic behaviour of employees. By implementing an accounting system, owners are able to leverage an essential in-built business function of providing adequate controls to safe guard organisational assets. An accounting system provides a means of implementing and improving the internal control structure of an organisation (Romney and Steinbart 2009). An effective accounting system provides an audit trail that allows frauds to be discovered and makes concealment difficult. Potential fraud can be discovered in accounting records by examining transactions that are anomalous or appear otherwise unreasonable.

Implementing a well-designed internal control policy enables an organisation to reduce opportunities for employees to commit occupational fraud. Further reduction in fraud may be achieved by introducing proactive fraud detection mechanisms that use computer-based technology (Broady and Roland 2008) to monitor and analyse business processes at an *"unprecedented level of detail"* (Alles et al. 2006 p.138).

1.1. Research problem

Fraud within organisations is a multi-billion dollar industry. Consequently, it is of major concern to industry and government (Goode and Lacey 2011 ; Best et al. 2009). Fraud costs the Australian economy approximately \$3 billion annually, and its frequency and financial impact continues to grow (Standards Australia 2008 ; KPMG 2008). Many organisations are poorly prepared to prevent and detect fraud (KPMG

2009). Fraud prevention is not perfect, therefore fraud detection is crucial. Fraud detection strategies are intended to quickly and efficiently identify those frauds that have circumvented preventative measures so that an organisation can take appropriate corrective action (Standards Australia 2008).

A review of various fraud surveys reveals that fraud is a crisis that is being faced by organisations internationally. Of all frauds detected in organisations, only 17% were attributed to the internal audit function (PwC 2009). According to PwC, internal audit is the primary method of detecting frauds, however the trend is that fewer frauds are being consistently detected. Opportunities to commit fraud are increasing, yet insufficient resources are being deployed to improve internal controls. Many organisations are considering the use of data analytics and information technology (IT) to detect fraud (KPMG 2008). Using IT to proactively detect fraud enables organisations to monitor and analyse large transaction datasets in real or near real time, a task that cannot be accomplished, practically, if done manually (Alles et al. 2006).

A study of the literature reveals that given the pervasiveness of enterprise systems additional research is necessary to advance the awareness, relevance, and practicality of continuous detection of potential fraud that uses technology to rapidly analyse large sets of transaction data (Kuhn Jr and Sutton 2010 ; Singleton and Singleton 2007 ; Debreceeny et al. 2005). It appears that prior research on continuous auditing does not appear to deliver a model that facilitates proactive and continuous monitoring for potential fraud without difficulties. Research is required to develop approaches of continuous auditing that are specifically applicable to auditing of

financial transactions in enterprise systems (Kotb and Roberts 2011 ; Debreceeny and Gray 2010 ; Kuhn Jr and Sutton 2010).

An issue often raised in the literature relates to information overload from alerts when implementing continuous fraud detection systems (Alles et al. 2008 ; Alles et al. 2006 ; Kuhn and Sutton 2006 ; Hunton et al. 2004). A related issue deals with the integrity of data used for continuous fraud detection (Kuhn Jr and Sutton 2010). These are important issues, as there appears to be potential demand for efficient and effective implementation of continuous fraud detection in organisations.

The modern global digital economy has significantly changed the way business is conducted and therefore the traditional approach to auditing can no longer be of real value to business performance or regulatory compliance. Most organisations conduct their business activities online and in real-time. This necessitates continuous monitoring and auditing thereby enabling internal auditors to perform their analyses of key business systems in real- or near real-time (Kotb and Roberts 2011 ; Kuhn Jr and Sutton 2010 ; Alles et al. 2006 ; Coderre 2005 ; Alles et al. 2002 ; Rezaee et al. 2002 ; Kogan et al. 1999).

The fraud landscape is dynamic, fast-moving and ever changing. Fraudsters are becoming more sophisticated in their use of technology and in their ability to commit and conceal fraudulent activities. As a result, fraud detection techniques must continue to evolve (Gill 2009 ; Singleton and Singleton 2007). With the advent of the digital economy and global market place employees now need only push a few keys on a computer keyboard to misdirect payments of vendor invoices, misplace

company assets or bribe suppliers. Physical possession of stolen property is no longer required and it is just as easy to program a computer to embezzle \$1 as it is \$1 million. Consequently, the best way to reduce the incidence of fraud is to implement fraud prevention and detection mechanisms. Industry intelligence (by means of Google searches and literature review) indicates that several accounting systems have integrated fraud prevention mechanisms built-in to software. However, there appears to be a lack of generic tools that proactively detect potential fraud in enterprise systems. The focus of this research is therefore on developing a generic model for proactive detection of potential fraud in enterprise systems. Actual fraud occurring can only be confirmed once potential fraudulent activities are fully investigated. Thus, the primary research question for this study is:

**Can a generalised model for proactive detection of potential fraud
in enterprise systems be developed?**

1.2. Study design

This research aims to answer the question whether *"a generalised model for proactive detection of potential fraud in enterprise systems can be developed"*. The research methodology for this study consists of the following separate yet interdependent stages.

- i). Literature review – to recognise theories and concepts that underpin this research and to identify gaps in the literature.
- ii). Create a catalogue of fraud symptoms (critical combinations and known fraud symptoms).

- iii). Identify data requirements to detect fraud in enterprise systems, in general and SAP, in particular.
- iv). Design, develop and implement prototype software on a stand-alone computer system.
- v). Perform experiments with simulated test data and case study data to verify program functionality of the prototype.
- vi). Seek support from experts for validity of the prototype.

The primary objective of this research is to explore and develop innovative methods for proactively detecting potential fraud in enterprise systems. The intention is to build a model for detection of potential fraud based on analysis of patterns or signatures¹. This research proposes a methodology for proactive detection of potential fraud that exploits audit trails in enterprise systems. The concept is demonstrated by developing a prototype. The aim of the prototype is to confirm the feasibility of implementing proactive detection of potential fraud in practice. The prototype is a software application that analyses transaction data from an SAP enterprise system for indicators of potential fraud. Reports and visualisations highlighting anomalous activities are produced. Further investigation of these findings may be initiated at the discretion of an auditor.

1.3. Key definitions

(In alphabetical order)

To ensure that terms used throughout the thesis are understood, they are defined here.

¹ The term 'pattern or signature' is used interchangeably throughout the thesis. This term refers to 'known or pre-existing' sequence of activities that may be used to help detect potential fraud. Activities performed by users' in an enterprise system may be recognised as anomalous by comparing them to these 'known patterns'.

Audit trails – provide a record of users' activities within an information system. They provide a means to accomplish several security related goals such as review of access, review of changes in security, review of attempts to bypass security and fraud detection. For the purpose of fraud detection they can be used to identify 'red flags' or anomalous activities perpetrated by real users acting in their own name, acting in collusion with other users, or by real users masquerading as others. In each case, the actions of these 'users' are recorded in audit trails (Albrecht et al. 2009).

Asset misappropriation – includes schemes in which perpetrators steal or misuse organisational resources, for example, skimming cash receipts, falsifying expense reports, shell company schemes, or payments to non-existent or ghost employees (ACFE 2010).

Billing schemes – perpetrators use false documentation, such as invoices, purchase orders or credit cards, to cause their employer to issue payments for some fraudulent purpose. Disbursement of funds is performed by an organisation in the same manner as a legitimate disbursement. The key to this scheme is the ability of a perpetrator to deceive an employer into willingly and unwittingly making a bogus payment (ACFE 2010).

Dashboard - a user interface that organises and presents information in a way that is easy to read. The aim is to integrate information from multiple sources into a unified display. For example, a product might obtain data from the local operating system in a computer, from one or more applications that may be running, and from one or more remote sites on the web and present it as though it all came from the same

source. Dashboards may be customised in a multitude of ways and named accordingly, for example the fraud analytics dashboard developed in this research organises and presents data about various indicators in the accounts payable system (Marane 2008).

Data - qualitative or quantitative attributes of a variable or set of variables. Data are typically the results of measurements and can be the basis of graphs, images, or observations of a set of variables. Data are often viewed as the lowest level of abstraction from which information and then knowledge are derived. Raw data or unprocessed data refers to a collection of numbers, characters, images or other outputs from devices that collect information to convert physical quantities into symbols. Data on its own carries no meaning. For data to become information, it must be interpreted and take on a meaning (Oxford 2012). (*Also see information*).

Data analytics - is a process of examining raw data with the purpose of drawing conclusions about that data. It is used in many industries to allow organisations to make better business decisions. Data analytics focuses on inference, to derive conclusions based on what is already known by the researcher. Banks and credit cards companies, for example, may analyse withdrawal and spending patterns to prevent fraud or identity theft. In the context of this research, analytics involves acquisition and analysis of electronic data to identify transactions that are anomalous or appear otherwise unreasonable. Data analytics often use dashboards that involve dynamic analysis and reporting, of real- or near real-time data obtained from a system (Nigrini 2011 ; Marane 2008 ; NIST 2003).

Embezzlement – an employee wilfully takes company's money or property by reason of employment or position of trust. Embezzlement may be direct or indirect. Direct embezzlement involves theft of company cash, inventory or other assets. Indirect embezzlement occurs when an employee establishes a shell corporation and issues false invoices to their employer for payment of goods and/or services that are not actually delivered (Albrecht et al. 2009 ; Wells 2008).

Enterprise systems – also referred to as enterprise resource planning (ERP) systems, integrate internal and external management information across all areas of an organisation. Business areas may include financial, accounting, manufacturing, sales, customer relationship management, human resources and so on. These systems facilitate flows of information within organisational boundaries and manage connections to external business partners (Kamhawi 2008 ; Koch and Wailgum 2008 ; Norris et al. 2000).

Information – as a concept has a diversity of meanings, from everyday usage to technical settings. In its most restricted technical sense it is an ordered sequence of symbols that can be interpreted as a message. Information can be recorded as signs, or transmitted as signals. It is any kind of event that affects the state of a dynamic system. Conceptually, information is the message (utterance or expression) being conveyed. The concept of information is closely related to notions of constraint, communication, control, data, form, instruction, knowledge, meaning, mental stimulus, pattern, perception, and representation (Oxford 2012). (*Also see data*).

Internal controls – are controls within an organisation that ensure data is processed correctly. They perform three important functions. Preventative controls deter problems before they arise. Detective controls discover problems as soon as they arise. Corrective controls remedy problems that have been discovered. Internal controls are an integral part of any organisations operating activities. They provide reasonable, rather than absolute assurance because providing complete assurance is difficult to achieve and prohibitively expensive (Romney and Steinbart 2009).

Materiality – the importance or significance of an amount, transaction, or discrepancy. The objective auditing financial statements is to enable an auditor to express an opinion whether financial statements are prepared in conformity with an identified financial reporting framework such as Generally Accepted Accounting Principles (GAAP). The assessment of what is material is a matter of professional judgment. Information is considered material if its omission or misstatement could influence the economic decision of users taken on the basis of financial statements (CPA 2009).

Middleware – is a software layer that provides a link between separate software applications'. It provides a common set of services that mediate interaction between these applications and computing resources. Initially middleware was intended to link newer applications to older systems, however, it also used to facilitate connections to multiple applications over computer networks. Organisations frequently use middleware to link data from databases distributed across an enterprise (Krakowiak 2007 ; Bouguettaya et al. 2006).

Module – in software, a module is part of a larger program. Modules are interchangeable components that perform specific functions. An individual module contains everything needed to accomplish a specific function. Modules may be integrated into larger programs through interfaces (Velaštin 1991).

Ponzi scheme – a type of fraudulent investment scheme that pays investors returns from their own money or monies paid by subsequent investors, rather than from profits earned by an individual or organisation running an investment scheme. The purpose of the scheme is to entice new investors to hand over funds by offering high short-term returns (Albrecht et al. 2009).

Red-flags – early warning symptoms or indicators of fraud. These symptoms may include changes in an employee's lifestyle, a general ledger being out of balance, an employee behaving suspiciously, or an anonymous tip that fraud is occurring. Investigation of these fraud symptoms may result in early detection of frauds (Albrecht et al. 2009).

Segregation of Duties – separating of accounting functions of authorisation, custody, and recording so as to minimise an employee's ability to commit and conceal fraud. Authorisation relates to approval of transactions. Recording relates to preparation of source documents, data entry and preparing of reconciliations and reports. Custody relates to handling cash or other assets, receiving and issuing payments (Romney and Steinbart 2009).

Shell corporation – a company which serves as a mechanism for business transactions without itself having any significant assets. They are not in themselves illegal and have legitimate business purposes; however, fraudsters use them to commit fraud. Fraudsters create these fictitious entities and submit false invoices, in the name of the entity, to an employer for payment (Wells 2008).

Validation - is an attempt to ensure that the right product is built and that it fulfils its specific intended purpose (IEEE 2004).

Vendor fraud - includes a broad range of schemes, from perpetrators that create fictitious shell companies and submit invoices for payment, to trusted suppliers that overcharge an organisation for more than is actually supplied or done. Some vendors may even collude with an organisation's own employees to help commit the fraud as part of a kickback scheme. An example of a non-accomplice vendor fraud scheme is when a perpetrator changes vendor payment details to a personal account, submits invoices for payment, and subsequently changes payment details back to the original values, causing payment to be misdirected to a personal bank account (Best et al. 2009 ; Wells 2002a).

Verification - is an attempt to ensure that a product is built correctly and that outputs of activities meet specifications imposed on them during the design phase (IEEE 2004).

Visualisation - is a general term used to describe any technology that enable users to 'see' data in order to help them better understand and put it in an appropriate context.

Visualisation tools go beyond standard charts and graphs, displaying data in more sophisticated ways such as dials and gauges, heat maps, tree maps and detailed bar and pie charts. Visualised data is frequently displayed in dashboards (TechTarget 2010).

White collar crime – a crime committed by a respectable person in a high position in an organisation during the course of their occupation. An individual personally benefits from the crime to the detriment of an organisation (Wells 2008).

1.4. Delimitations of scope

There is some delimitation of scope of this study to ensure that it concentrates on the research questions, propositions and objectives only. When considering an automated solution for proactive fraud detection, the focus has to be on questions that can be answered with the aid of computerised tools (Lanza 2007). Some questions are too subjective, for example, "Are the vendor's goods or services of good quality?" Any effort to develop an automated solution will require evidence that is documented in an enterprise system's audit trails and that can be investigated using data analytics tools. Transactions that occur outside an enterprise system cannot be investigated using this methodology.

The Association of Certified Fraud Examiners (ACFE) classifies occupational fraud into three broad categories; asset misappropriation, corruption and fraudulent statements. Several fraud surveys have found asset misappropriation to be the most common category of fraud perpetrated by non-management employees (ACFE 2010)

; KPMG 2009 ; PwC 2009). Cash assets were more frequently targeted than non-cash assets and billing schemes were the most common method used to misappropriate cash assets. This research examines the feasibility of developing a model for proactive detection of potential billing fraud schemes involving shell companies and non-accomplice vendors within accounts payable. Hereinafter these schemes are referred to as ‘vendor frauds.’

Large scale implementations of enterprise systems have resulted in many organisations being highly automated and fully integrated. The development of this enterprise system environment provides the necessary infrastructure for the effective evolution of the auditing function from a periodic event to an ongoing process through the use of computer-based technology. Enterprise systems software are available from several vendors, including SAP, Oracle and Microsoft, and collectively has 71% of market share world-wide. For several years, however, Germany-based enterprise software company SAP has consistently been the market leader (SAP 2010 ; Lager and Tsai 2008). In 2010 Gartner (2010) recognised SAP as the leading vendor of enterprise systems software accounting for 22% of the market. Many organisations have realised that SAP solutions are important to their success. Several Fortune 500 companies, including IBM, Toyota, Apple, Coca-Cola, and Google use SAP exclusively for their core day to day operations including accounting and financial applications, procurement, order processing and supplier management, inventory management, and HR management and payroll functions (CMU 2011 ; Gartner 2010 ; BOS 2009). The prototype software being developed by this research exploits SAP audit trails for proactive detection of potential vendor

fraud schemes. [Testing and evaluation of the prototype will provide evidence that the concept of proactive detection of potential fraud is feasible in practice.]

The prototype developed in this research is intended to address the primary research question by providing evidence that "*a generalised model for proactive detection of potential fraud in enterprise systems can be developed*". Such a prototype is meant to demonstrate that the "*concept of proactive detection of potential fraud*" is feasible in practice. It is a limited version meant for showcasing the concept and for testing purposes only. Some functions may be incomplete, not implemented or may not even work at all.

The prototype makes no assumption about individual SAP installations. It relies exclusively on transaction data obtained from the financial module to perform its analysis. Other factors that are not considered by the prototype include; sophistication of fraudsters, posting of transactions by system and security administration staff, collusion between fraudsters to circumvent internal controls, and organisations wherein segregation of duties may not be feasible due to small numbers of staff. These situations require additional compensating manual processes to safeguard against inappropriate activities.

The scope of this study is therefore limited to detection of potential vendor fraud schemes involving shell companies and non-accomplice vendors in an SAP enterprise system using prototype software developed for this purpose. The study makes no claims to be able to identify any 'actual' fraudulent activities but is limited to extracting data that provide symptomatic evidence that fraudulent activities might

have occurred, Throughout this thesis the term 'fraud', 'fraud detection', or 'fraud detection tool' means 'potential' fraud not 'actual' fraud.

1.5. Research significance

Australia has an estimated \$3 billion per year financial fraud problem that continues to worsen (Standards Australia 2008). This research directly addresses the national research priority of safeguarding Australia, in particular the priority goals of critical infrastructure and protecting Australia from terrorism and crime (ARC 2011). Australian criminal codes define fraud as 'crime' and prescribe prison terms of up to ten years where fraud is committed by an employee or company director.

Although fraud surveys reveal that fraud is increasing, it is difficult to know for sure. It is impossible to know what percentage of perpetrators are caught. There may be frauds that are never discovered. Many frauds that are detected are handled quietly by the victim organisations as they are more concerned about reputation, and costs associated with fraud investigations

Enhancing the ability of organisations in the private and public sectors to detect fraud is important for the following reasons:

- Detecting financial fraud will enhance the stability of businesses and the economy. Extraordinary impacts have been observed on stock exchanges both in the United States and Australia following major financial frauds (for example, Enron, WorldCom, HiH Insurance and Lehman Brothers).
- Losses incurred from fraud reduce a firm's income on a dollar-for-dollar basis. This means that for every \$1 of fraud, net income is reduced by \$1. If a firm's

profit margin at the time of the fraud was ten percent, the firm would have to generate ten times more revenue to restore the effect on net income.

- Outcomes of this research will enhance protection for stakeholders (shareholders, lenders, and employees) that stand to lose a great deal when major financial frauds are detected.
- This research is of particular relevance to financial institutions and the retail and public sectors (for example, *Queensland Health Hohepa Morehu-Barlow Fraud Case*) as they appear to be major financial fraud targets in Australia. These organisations represent a major component of the nation's critical infrastructure.

This research extends prior research that focuses principally on fraud prevention rather than its detection (Goode and Lacey 2011 ; Albrecht et al. 2009 ; Coderre 2005 ; Best 2005). Preventative fraud controls are intended to reduce or eliminate opportunities to perpetrate fraud by: i) implementing segregation of duties; ii) having a system of proper authorisations; iii) implementing physical safeguards; iv) implementing independent system checks; and v) having audit trails (COSO 1992). Once a system is violated, detective controls may help in identifying the occurrence of harm (Abu-Musa 2007). The objective of this research is therefore, to explore and develop innovative methods to proactively detect potential fraud by continuous monitoring and analysis of audit trail data in enterprise systems. Continuous monitoring enables an auditor to provide a degree of assurance on information shortly after disclosure (Rezaee et al. 2002). This enables an organisation to quickly and efficiently identify activities that circumvent preventative measures and take

appropriate corrective action thereby reducing the propensity for losses associated with future fraud.

1.6. Structure of dissertation

Chapter 1 - Introduction. This first Chapter discusses the background to this study which includes the research problem, study design, definitions of key terms used throughout this dissertation, delimitation of the scope, and significance of this study.

Chapter 2 - Literature review. This Chapter discusses the relevant literature on fraud, its theoretical underpinnings that constitute this research and its detection in enterprise systems. Previous studies in this area of research are discussed and relevant gaps in the literature are identified. The primary research question is developed from the gaps identified in the literature.

Chapter 3 - Research methodology. Based on the literature review, a theoretical framework is developed and discussed in this Chapter. The research methodology used during the course of this study for data collection, methods to detect fraud, and analysis are also discussed. An expert panel protocol and instrument for collecting evidence for validation of the prototype are also developed.

Chapter 4 - Prototype design. This Chapter addresses research propositions RP1a, RP1b and RP1c. The SAP enterprise system is investigated to determine whether it documents adequate data in its audit trails to allow retrospective monitoring of user activities. A conceptual design of a prototype is proposed. Detailed design

specifications are produced. A logical design and detection algorithms are subsequently developed from the design specifications.

Chapter 5 - Prototype implementation and testing. This Chapter addresses research propositions RP2a, RP2b and RP2c. Implementation and test results of the prototype are described with reference to a number of appendices. Verification of the prototype is achieved by performing a series of tests using test data involving simulated activity. Case study data from a large international manufacturing company is processed using the prototype, exposing it to live data. Validation is achieved by obtaining independent reviews from auditing practitioners and an expert panel demonstration. Timing experiments are conducted to provide evidence in support of auditor productivity.

Chapter 6 - Conclusion and future research. The final Chapter summarises conclusions of this study, contributions to the literature and auditing practice, limitations, and recommendations for future research and extensions to the prototype.

1.7. Conclusion

Given the pervasiveness of enterprise systems there is limited research regarding proactive detection of potential fraud in enterprise systems. There appears to be no research in developing a generalised model for proactive detection of potential fraud in enterprise systems. Additional research is necessary to advance the awareness, relevance, and practicality of continuous fraud detection in enterprise systems. Limited research has been conducted in the use of embedded audit modules within enterprise systems, however, it does not appear to deliver a model that facilitates

proactive and continuous monitoring for potential fraud without difficulties. Research is required to develop innovative approaches for proactive detection of potential fraud, and to demonstrate how this can be done efficiently and effectively. Information overload from alerts produced by automated fraud detection systems also appears to be a problem.

These are important issues, as there appears to be potential demand for efficient and effective implementation of proactive detection of potential fraud in organisations. The research conducted in this study will make a contribution to the literature by seeking to address these gaps, and to enhance the body of knowledge in the area of proactive detection of potential fraud in enterprise systems.

CHAPTER 2

Literature Review

2.0. Introduction

Fraud is inherent in all organisations. Edwin H. Sutherland, a criminologist at Indiana University, coined the phrase "*white-collar crime*" in 1939 (Sutherland 1940). Donald R. Cressey, a student of Sutherland, was especially interested in embezzlers, whom he referred to as "trust violators". He was intrigued by what led these people to be overcome by temptation. Upon completion of his work, Cressey developed the classic model for the occupational offender. This model, more commonly referred to as the 'fraud triangle', underpins the theoretical foundation for this study (Cressey 1950).

According to the ACFE's Report to the Nations on Occupational Fraud & Abuse, "a typical organisation loses five percent of its annual revenue to fraud. Applied to the estimated 2009 Gross World Product of \$58.07 trillion, this figure translates to a potential total fraud loss of more than \$2.9 trillion" (ACFE 2010 p.4). Consequently, it is of major concern to industry and government (Goode and Lacey 2011 ; ACFE 2010 ; Best et al. 2009). Fraud costs the Australian economy approximately \$3 billion annually (Standards Australia 2008), and its frequency and financial impact continues to grow (Standards Australia 2008 ; KPMG 2008). Many organisations are poorly prepared to prevent and detect fraud (KPMG 2009 ; KPMG 2008 ; KPMG

2007 ; KPMG 2004). Fraud prevention is not perfect therefore, fraud detection is crucial. Fraud detection strategies are intended to quickly and efficiently identify those frauds that have circumvented preventative measures so that an organisation can take appropriate corrective action (Standards Australia 2008).

A review of various fraud surveys revealed that fraud is a crisis that is being faced by organisations internationally. Of all frauds detected in organisations, only 17% were attributed to the internal audit function (PwC 2009). According to PwC, internal audit was the primary method of detecting fraud, however the trend was that fewer frauds are being consistently detected. With advances in information technology and emergence of electronic business, modern enterprise systems may record millions of transactions annually. An auditor may extract a small sample of these during a financial audit. Suppose a fraudster perpetrates only a few frauds annually, it is plausible that none of them may be discovered by the financial audit. Many fraudsters rely on this to conceal fraud. Thus, while opportunities to commit fraud continue to increase, it appears that insufficient resources are being deployed to improving internal controls. Many organisations are considering the use of data analytics and information technology (IT) to detect fraud (KPMG 2008). Using IT to proactively detect potential fraud enables organisations to monitor and analyse large transaction datasets in real- or near real- time (Edge and Falcone Sampaio 2009 ; Alles et al. 2006), a task that cannot practically be accomplished by an internal auditor. A study of the literature reveals that there is a need for further research into proactive detection of potential fraud that uses technology to rapidly analyse large sets of transaction data (Kotb and Roberts 2011 ; Kuhn Jr and Sutton 2010 ; Debreceeny and Gray 2010). This research aims to explore and develop innovative

methods of using information technology to proactively detect potential fraud in enterprise systems. A fundamental first step in studying the fraud problem is to unambiguously define fraud itself.

2.1. Definition of fraud

There are two principal methods of getting something from others illegally. They can either be physically forced, or they can be deceived into giving up their assets. The first type is called robbery and the second is fraud. Albrecht et al. (2009) defines fraud as a deception made for personal gain. "Deception" is key. The most common definition of fraud according to Webster's Dictionary (2001 p.380) is:

"Fraud is a generic term that embraces all the multifarious means which human ingenuity can devise, which are resorted to by one individual, to get an advantage over another by false representations. No definite and invariable rule can be laid down as a general proposition in defining fraud, as it includes surprise, trickery, cunning and unfair ways by which another is cheated. The only boundaries defining it are those which limit human knavery."

Australian Government *Fraud Control Guidelines* define fraud as (ComLaw 2011 p.4):

"theft; accounting fraud (false invoices, misappropriation etc); unlawful use of, or obtaining property, equipment, material or services; causing a loss, or avoiding and/or creating a liability; providing false or misleading

information, or failing to provide it when there is an obligation to do so; misuse of assets, equipment or facilities; making, or using false, forged or falsified documents; and wrongfully using information or intellectual property."

Furthermore, the Government identifies fraud as targeting revenue, benefits, property, information and intelligence, funding and grants, entitlements, facilities, and money or property. Benefits obtained fraudulently are not restricted to monetary or material benefits, and may be tangible or intangible, including unauthorised provision of access to or disclosure of information. Benefits may also be obtained by third parties in addition to the fraud perpetrator.

Two types of fraud can be distinguished: misappropriation of assets and fraudulent financial reporting (Casabona and Grego 2003 ; ASB 2002). Misappropriation of assets is often referred to as 'employee fraud' and involves theft of an organisation's assets. Fraudulent financial reporting involves deliberate misstatements or omissions of amounts or disclosures of financial statements to deceive investors and creditors, increase share price, meet cash flow needs or hide company losses and problems (Romney and Steinbart 2009 ; Wells 2008 ; Casabona and Grego 2003). The Association of Certified Fraud Examiners (ACFE) extends this definition by classifying frauds and the methods used to commit them. The term fraud is therefore used to describe a wide variety of crimes and swindles that range from Ponzi schemes, identity and data theft to falsification of financial reports.

The ACFE (2010 p.6) defines occupational fraud as:

"...the use of one's occupation for personal enrichment through the deliberate misuse or misapplication of the employing organization's resources or assets..."

Occupational fraud is very broad and it encompasses a range of transgressions by employees at all levels of an organisational hierarchy. Occupational fraud can be divided into three categories: fraudulent statements, corruption and asset misappropriation (ACFE 2010) (Figure 2.1).

The three broad categories of occupational fraud generally correspond to the three broad levels of hierarchy in an organisation. The production of fraudulent statements is often orchestrated at board/executive-level, while corruption mostly affects people in management positions, i.e. people who have the power to influence company's decisions in favour of the corrupting party. Asset misappropriation is most common among non-management employees, primarily because these employees do not have opportunities to commit fraud at the previous two levels (ACFE 2010 ; KPMG 2010 ; Albrecht et al. 2009 ; Wells 2008).

2.1.1. Asset misappropriation

Asset misappropriation is the most common category of fraud (ACFE 2010). This is consistent with studies that show fraud by non-management employees is most common (ACFE 2010 ; KPMG 2010 ; KPMG 2009 ; KPMG 2008). This is, primarily because most organisations employ more non-management than management employees.

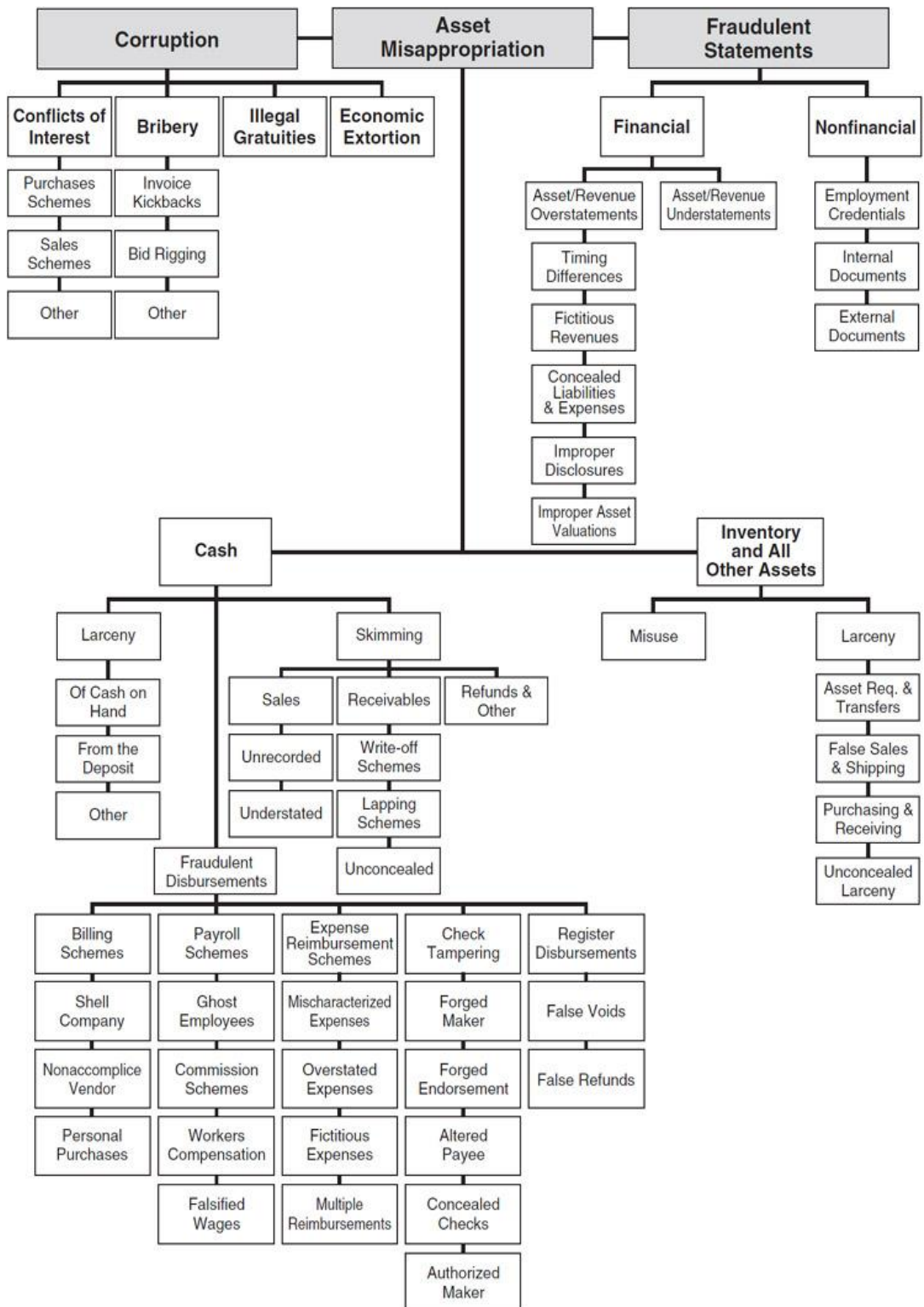


Figure 2.1: Categories of occupational fraud

Source: (ACFE 2010)

Asset misappropriation involves the misuse and appropriation of company assets for personal gain. It is divided into two subcategories: a) misappropriation of cash, and b) misappropriation of inventory and other assets that can usually be turned into cash. Cash schemes involve larceny, skimming and fraudulent disbursements. Non-cash schemes involve misuse and larceny (ACFE 2010 ; Albrecht et al. 2009 ; Coenen 2008 ; Wells 2008 ; Potla 2003).

Larceny involves taking an employer's cash or other assets without the consent and against the will of the employer, after it has been recorded in the company's records. The two main methods are: a) theft of cash on hand, and b) theft from the cash bank deposit. In order to prevent detection, the fraudster will have to create or modify accounting documents explaining the cash shortage. Skimming schemes are schemes where cash is stolen before a book entry is made. This may involve not recording or understating of sales. Receivables schemes involve write-off and lapping schemes. In write-off schemes an employee collects money for receivables but writes off the receivables instead of recognising them as paid. In lapping schemes an employee steals one client's payment and later covers it by paying their account with another client's payment. This type of fraud is difficult to perpetrate as it requires ongoing maintenance. Fraudulent disbursements are methods where the misappropriation of funds appears to be for legitimate business events. There are five groups under fraudulent disbursements i.e. billing schemes, payroll schemes, expense reimbursement schemes, and cheque tampering and register disbursements.

Billing schemes involve making payments for inaccurate or false expenses or payments for personal purchases. The aim of false or inaccurate payments is to

redirect the money back to oneself. Billing schemes may involve the use of a shell company (i.e. a company created for the sole purpose of committing fraud) which submits fictitious invoices for payment. The fraudster must be able to influence approval of vendors and invoices. Non-accomplice vendors may also be used by intercepting payments to them, intercepting refunds or by stealing legitimate payments made to them. Purchasing (credit) cards are another method for making personal purchases at the company's expense. Payroll schemes may involve ghost employees being entered into the system. This scheme is similar to shell vendors. Alternatively, employees may falsify the amount of hours they work or inflate the commission they should be paid for sales. Another method involves claiming compensation for fictitious injuries.

Expense reimbursement schemes involve submitting false expenses and then being reimbursed for them. Methods include expense schemes, overstated expenses, fictitious expenses, and multiple reimbursement schemes. In cheque tampering schemes a perpetrator is actually involved in the issuing of a cheque. The signature of the person/s that normally approves a cheque is forged. Another method is to claim to be the party written on a legitimate cheque or to change the payee written on the cheque. In the concealed cheque technique the perpetrator submits a fraudulent cheque amongst legitimate cheques so that the cheque will be 'rubber-stamped' signed. In authorised marker schemes a fraudster is the person who can approve and issue cheques. This situation makes it quite easy to issue cheques to a perpetrator as it relies on weak and ineffective controls. In register disbursement schemes cash is removed from a cash register and recorded on the system using techniques such as false voids or false refunds. This kind of fraud occurs in cash

businesses such as retail, restaurants or bars where employees may void an order provide customers with ordered goods and keep monies for themselves.

Non-cash assets such as inventory and equipment may be misappropriated in a number of ways that can range from taking home a box of pens to theft of millions of dollars of property. Company assets may be misused ('borrowed') or stolen. Assets such as company vehicles, supplies, computers and other office equipment may be misused by employees to do personal work during company time. Costs of non-cash asset misuse are difficult to quantify. Company assets may also be stolen. Employees may create false documentation to ship company merchandise to personal addresses, or they may simple take company assets without trying to account for their absence.

2.1.2. Corruption

While misappropriating assets sees a physical reduction of assets, such as cash, corruption schemes involve employees using their influence or official position in organisations to unlawfully obtain benefits for themselves or other persons, contrary to the rights of others (ACFE 2010 ; Wells 2008). Corruption schemes are divided into bribery, conflict of interest, economic extortion and illegal gratuity. Bribery involves offering something of value in order to influence a business decision. Illegal gratuities involve giving an employee something of value to reward a decision rather than to influence it. Economic extortion occurs when one person demands payment from another, for example an employee demands payment from a supplier before awarding them a contract. Conflicts of interest arise when an employee has an undisclosed economic or personal interest in a transaction that

adversely affects an organisation, for example the employee has a stake in a company that his employer is transacting with possibly through a family relationship i.e. spouse, uncle, cousin, and so on.

2.1.3. Fraudulent financial statements

Financial statement fraud is the intentional misstatement or omission of material information from an organisation's financial statements with the intention of deceiving investors and creditors (ACFE 2010 ; Wells 2008). Common schemes involve recording of fictitious revenues, concealing liabilities or expenses and artificially inflating reported assets.

This section provides a definition of fraud that is applicable to this research. According to various fraud surveys (ACFE 2010 ; KPMG 2010), asset misappropriations appear to be perpetrated most frequently. It is most common among non-management employees, primarily because these employees do not have opportunities to commit fraud at the previous two levels (ACFE 2010 ; KPMG 2010 ; Albrecht et al. 2009 ; Wells 2008). Consequently, the broad area of investigation for this research is '*Asset Misappropriation*' with a specific focus on '*Billing Schemes*' within accounts payable (AP). Fraud can occur through a variety of schemes that may be perpetrated by employees at all levels of an organisational hierarchy. Fraud is perpetrated in secret and concealed by an employee, this action results in a direct or indirect benefit to a perpetrator, while a victim organisation may suffer a loss of assets, revenue or business opportunity.

2.2. Occurrence and cost of fraud

The number and value of fraud incidents in Australia and New Zealand continues to increase significantly (ACFE 2010 ; KPMG 2008). Forty five percent of organisations experienced incidents of fraud between 2006 and 2008. The level of fraud suffered was higher in large organisations. Sixty-two percent of organisations with 1,000 to 10,000 employees experienced at least one fraud, while 89% of organisations employing more than 10,000 people experienced at least one fraud (KPMG 2008). The median loss suffered by organisations worldwide annually was \$139,000 for organisations with 1,000 to 10,000 employees, and \$164,000 for organisations with more than 10,000 employees. More than 40% of privately owned companies and more than 30% of publicly owned companies were victims of fraud (ACFE 2010).

Asset misappropriation is the most common type of fraud, occurring in more than 86% of all cases (ACFE 2010) (Table 2.1). The median loss from asset misappropriation was \$135,000. *(Note: the sum of percentages in Table 2.1 exceeds 100% because several cases involved schemes from more than one category).*

Table 2.1: Categories of occupational fraud and abuse

Category	% of all Cases	Median Loss
Asset Misappropriation	86.3%	\$135,000
Corruption	32.8%	\$250,000
Fraudulent Statements	4.8%	\$4,100,000

Source: ACFE (2010)

Asset misappropriation schemes involve the theft of cash and non-cash assets. Cash assets i.e. cash receipts, cash disbursements and cash on hand (such as petty cash and

cash in vault) were more frequently targeted (83.7%) than non-cash assets i.e. inventory, supplies, fixed assets, intellectual property, investments and proprietary information (16.3%)

Fraudsters use various schemes to misappropriate assets from their employees. The ACFE (2010) survey classified asset misappropriation into nine sub-categories i.e. cash larceny, skimming, billing, payroll, expense reimbursements, cheque tampering, cash register disbursements, cash on hand misappropriation and non-cash misappropriation. Of all cases reported in the United States of America, billing schemes occurred most frequently (26%).

Table 2.2: Sub-categories of asset misappropriation (US Data)

Category	% of all Cases	Median Loss
Billing	26.0%	\$128,000
Non-Cash Misappropriations	16.3%	\$90,000
Expense Reimbursements	15.1%	\$33,000
Skimming	14.5%	\$60,000
Cheque Tampering	13.4%	\$131,000
Cash on Hand Misappropriations	12.6%	\$23,000
Cash Larceny	9.8%	\$100,000
Payroll	8.5%	\$72,000
Cash Register Disbursements	3.0%	\$23,000

Source: ACFE (2010)

A billing scheme is any scheme in which an employee causes their employer to issue payment by submitting invoices for fictitious goods or services, inflated invoices or invoices for personal purchases (Wells 2008 ; Coenen 2008 ; ACFE 2010). The median loss suffered from billing schemes was \$128,000 (Table 2.2).A similar trend was observed in cases reported from Oceania (Australia, Fiji, Micronesia and New Zealand) (Table 2.3).

Table 2.3: Sub-categories of asset misappropriation (Oceania Data)

Category	% of all Cases
Non-Cash Misappropriations	30.0%
Billing	27.5%
Cheque Tampering	17.5%
Skimming	12.9%
Expense Reimbursements	10.0%
Cash on Hand Misappropriations	10.0%
Cash Larceny	7.5%
Payroll	5.0%
Cash Register Disbursements	2.5%

Source: ACFE (2010)

The main factor that contributed to fraudulent activities in an organisation was a lack of internal controls (37.8%) e.g. poor segregation of duties. The second highest contributing factor was overriding existing internal controls (19.2%) (Table 2.4).

Table 2.4: Control weakness that contributed to fraud

Category	% of all Cases
Lack of Internal Controls	37.8%
Override of Existing Internal Controls	19.2%
Lack of Management Review	17.9%
Poor Tone at the Top	8.4%
Lack of Competent Personnel in Oversight Roles	6.9%
Lack of Independent Checks/Audits	5.6%
Lack of Employee Fraud Education	1.9%
Lack of Clear Lines of Authority	1.8%
Lack of Reporting Mechanism	0.6%

Source: ACFE (2010)

Non-management employees were found to be the largest group of fraud perpetrators (42.1%). Managers perpetrated 41% of occupational frauds and owners perpetrated 16.9% (Figure 2.2). The median loss suffered from frauds perpetrated by non-management employees was \$80,000. This amount increased to \$200,000 for managers and to \$723,000 for owner-related frauds.

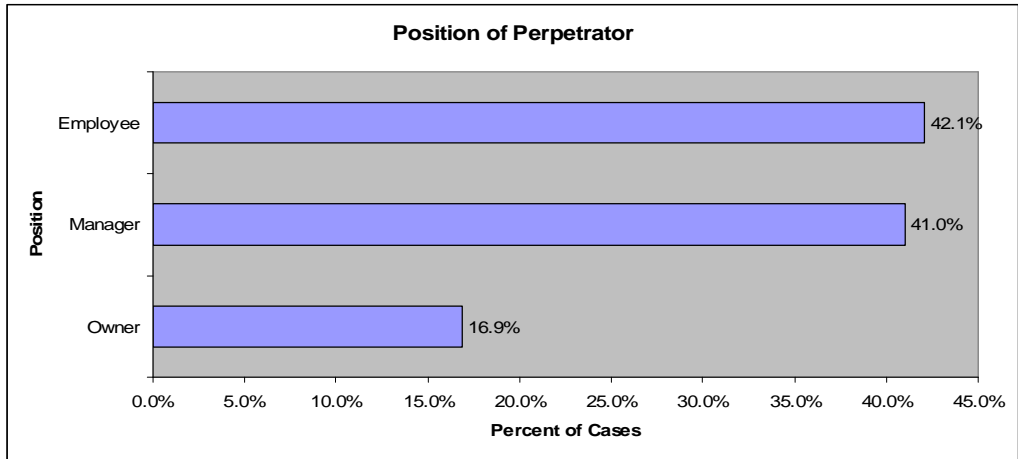


Figure 2.2: Position of fraud perpetrator

Source: ACFE (2010)

More than 80% of all frauds were committed by employees in six departments, namely accounting, operations, sales, executive/upper management, customer service and purchasing. The frauds in these six departments accounted for 95% of all losses (Figure 2.3).

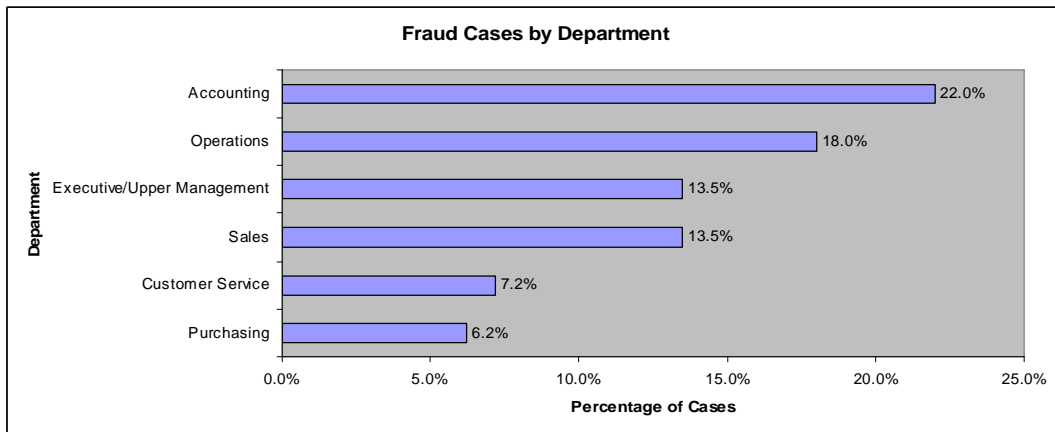


Figure 2.3: Fraud cases based on perpetrator's department

Source: ACFE (2010)

Among the six departments with the highest frequency of fraud cases, upper management (\$829,000) and purchasing (\$500,000) caused the highest median losses (Figure 2.4).

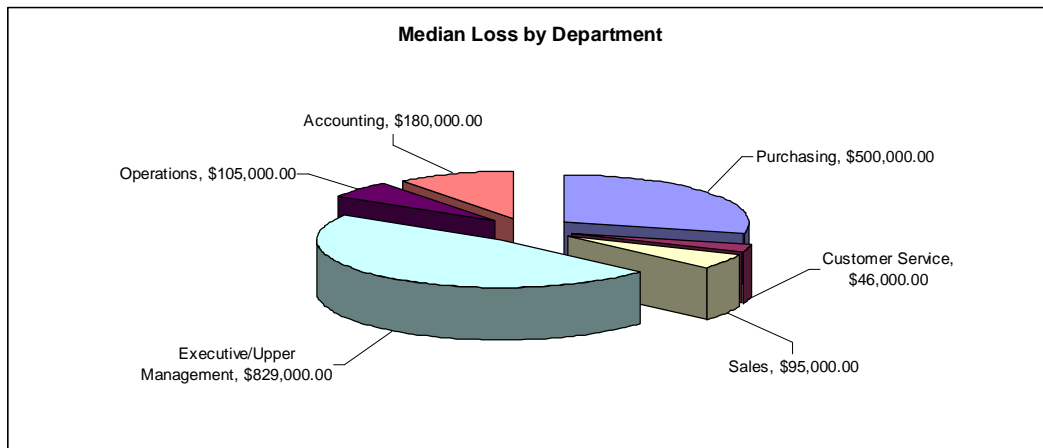


Figure 2.4: Median loss by perpetrator's department

Source: ACFE (2010)

The most common fraud schemes perpetrated in the accounting department were cheque tampering (33.2%) and billing fraud (30.8%). In the operations department corruption (30.8%) and billing fraud (22.1%) were the top two schemes. In the sales department it was corruption (33.8%) and theft of non-cash assets (23.6%). Executives mostly engaged in corruption (48.7%), billing (40.6%) and expense reimbursement schemes (29.9%). Corruption (21.7%), skimming (19.2%), theft of cash on-hand (18.3%) and fraudulent register disbursements (8.3%) were the top ranked fraud schemes perpetrated by customer service employees.

From the above analysis of the occurrence and cost of fraud in organisations, asset misappropriation is identified as the most common type of fraud perpetrated. Employees at all levels of an organisational hierarchy were involved in perpetrating frauds, however the largest group are non-management employees. These employees used billing schemes to perpetrate fraud against organisations. Two weaknesses exploited by fraudsters were the lack of internal controls and overriding of existing

internal controls. According to KPMG (2010), there was a 6% increase in poor internal controls as a contributing factor to fraud when compared to their previous fraud survey (KPMG 2008). Identifying the underlying reasons that motivate these individuals to deliberately violate their position of trust is essential in understanding the concept of fraud and its detection.

2.3. Motivation to commit fraud

Human needs can be satisfied by either honest or dishonest means. What is it that motivates an employee to choose dishonest methods to meet or satisfy their needs within an organisation? A factor in understanding human behaviour in organisations is why organisations exist in the first place.

Organisations exist because entrepreneurs identify and exploit market opportunities to create wealth (Coase 1937). As an organisation grows the entrepreneur can no longer manage the business alone. The roles of ownership and management need to be separate (Berle and Means 1932) with operational decisions becoming the responsibility of management (Jensen and Meckling 1976). With management dispersed, conflicts of interest may arise between management and the owner(s) leading to the 'principal-agent' problem (Fama and Jensen 1983 ; Jensen and Meckling 1976). In these situations management may use organisational resources to resolve conflicts in their favour (Demsetz 1983). If management is motivated by selfish and opportunistic behaviour, then owners interests will not be served (Berle and Means 1932).

Owners' abilities to effectively identify selfish and opportunistic behaviours by management are diminished because of information asymmetry i.e. the 'principal-agent' problem. Information asymmetry occurs as a result of agents (management) having more information about operational activities of an organisation than a principal (owner) (Fama and Jensen 1983 ; Jensen and Meckling 1976). Contractual relationships exist between principal and agent in an organisation (Adams 1994 ; Jensen and Meckling 1976). Both parties use this relationship to maximise their wealth. This means that agents with self-centred motives may violate their position of trust by acting against the interests of the principal (Adams 1994).

Behaviours of individuals in organisations are a set of complex social interactions which is explained by Frederick Herzberg's 'Theory of Motivation' (1959) and Abraham Maslow's 'Hierarchy of Needs' (1943). Herzberg's (1959) two factor theory (motivation-hygiene theory) examines factors that affect people's attitudes about work. Factors such as company policy, supervision, interpersonal relations, working conditions, and salary are hygiene factors i.e. factors that do not give positive satisfaction although their absence results in dissatisfaction. Conversely, motivators such as achievement, recognition, the work itself, responsibility, and advancement, determine job satisfaction. Some of his key findings were:

- i). people become dissatisfied by a poor environment, but they are seldom satisfied by a good environment;
- ii). preventing dissatisfaction is just as important as encouragement of satisfaction; and

- iii). hygiene factors operate independently of motivation factors; an individual could be highly motivated in his job yet dissatisfied with his work environment.

Maslow's (1954) 'Hierarchy of Needs'² describes how people satisfy various personal needs within contexts of their jobs. There is a general pattern of needs recognition and satisfaction that people follow in the same sequence. A person cannot recognise or pursue the next higher need in the hierarchy until their current need is satisfied (Maslow 1943). Individuals in an organisation commonly have difficulty expressing what they want from their job. Employers may impose conditions that they believe to be in the best interests of employees (Green 2000), leading to dissatisfaction among employees. They may seek relief and satisfy their needs by violating their position of trust and resorting to fraudulent behaviour.

Donald R. Cressey was interested in studying employees that violate their position of trust to satisfy to satisfy their needs. He was especially interested in identifying circumstances that led employees to be overcome by temptation. Cressey hypothesized that (Cressey 1953 p.30):

"Trusted persons become trust violators when they conceive of themselves as having a financial problem which is non-shareable, are aware this problem can be secretly resolved by violation of the position of financial; trust, and are able to apply to their own conduct in that situation verbalizations which enable them to adjust their conceptions of themselves

² Maslow's hierarchy of needs is represented by a pyramid, with the physiological need at the bottom, and the self-actualisation need at the top. The other three are safety, love and esteem, moving up the pyramid respectively.

as trusted persons with their conceptions of themselves as users of entrusted funds or property."

Cressey's work provided valuable insight into why people commit fraud and it led to the development of the 'fraud triangle' (Figure 2.5) (Cressey 1950). The three key elements of the fraud triangle are pressure (an un-shareable need), rationalisation (of personal ethics), and opportunity (lack of adequate controls and knowledge to commit a fraud). All three elements must be present in order for a fraud to be perpetrated (Albrecht et al. 2008 ; Cressey 1950).

Pressure is related to an employee's perceived immediate need for an asset due mainly to financial difficulties. This causes a person to take significant risks in order to obtain the desired resource. While it is not within the scope of an auditor's responsibilities to resolve issues leading to pressures associated with fraud, it is important to bring these situations to management's attention if uncovered. It is a reasonable assumption that pressure is not a factor that can be captured in transaction data within an enterprise system. It tends to be more a human condition for organisational behaviour and psychology researchers to study.

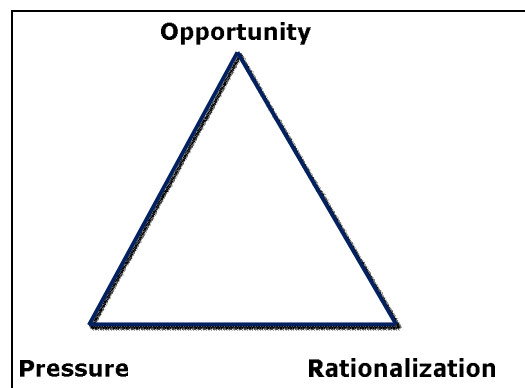


Figure 2.5: Fraud triangle

Source: (Albrecht et al. 2009)

Opportunities to commit fraud arise when an employee is in a position of trust, internal controls are weak or non-existent and when an employee has the applicable knowledge to commit a fraud. An employee perceives that an opportunity to commit fraud exists, commits it and conceals it. Good controls are important in limiting opportunities to commit fraud, but they are not fool proof. Flaws in internal controls provide opportunities for committing fraud however digital 'footprints' or 'signatures' of these activities are recorded in transaction data and audit trails within an enterprise system. These 'footprints' can be analysed in order to determine whether a fraud has potentially occurred. Interestingly, the ease with which a fraud can be perpetrated in a digital environment provides the tools to investigate and identify fraud i.e. computer technology.

Rationalisation or integrity is the third element of the fraud triangle. Individuals do not commit fraud unless it is consistent with their own personal code of ethics or belief. This limiting factor prevents most employees from committing fraud even though an opportunity may exist. When fraud is committed and detected most perpetrators rationalise their behaviour to match with their personal beliefs and/ or pressures they face. Again, this factor cannot be captured in transaction data within an enterprise system.

Several theories have been identified that inform and underpin this research (Figure 2.6). A common theme in each of these theories is that if a conflict of interest arises between owner(s) and employees, this may lead to dissatisfaction among employees. The affected employee(s) may seek relief by resorting to fraudulent behaviour when an opportunity presents itself. Each of the theories

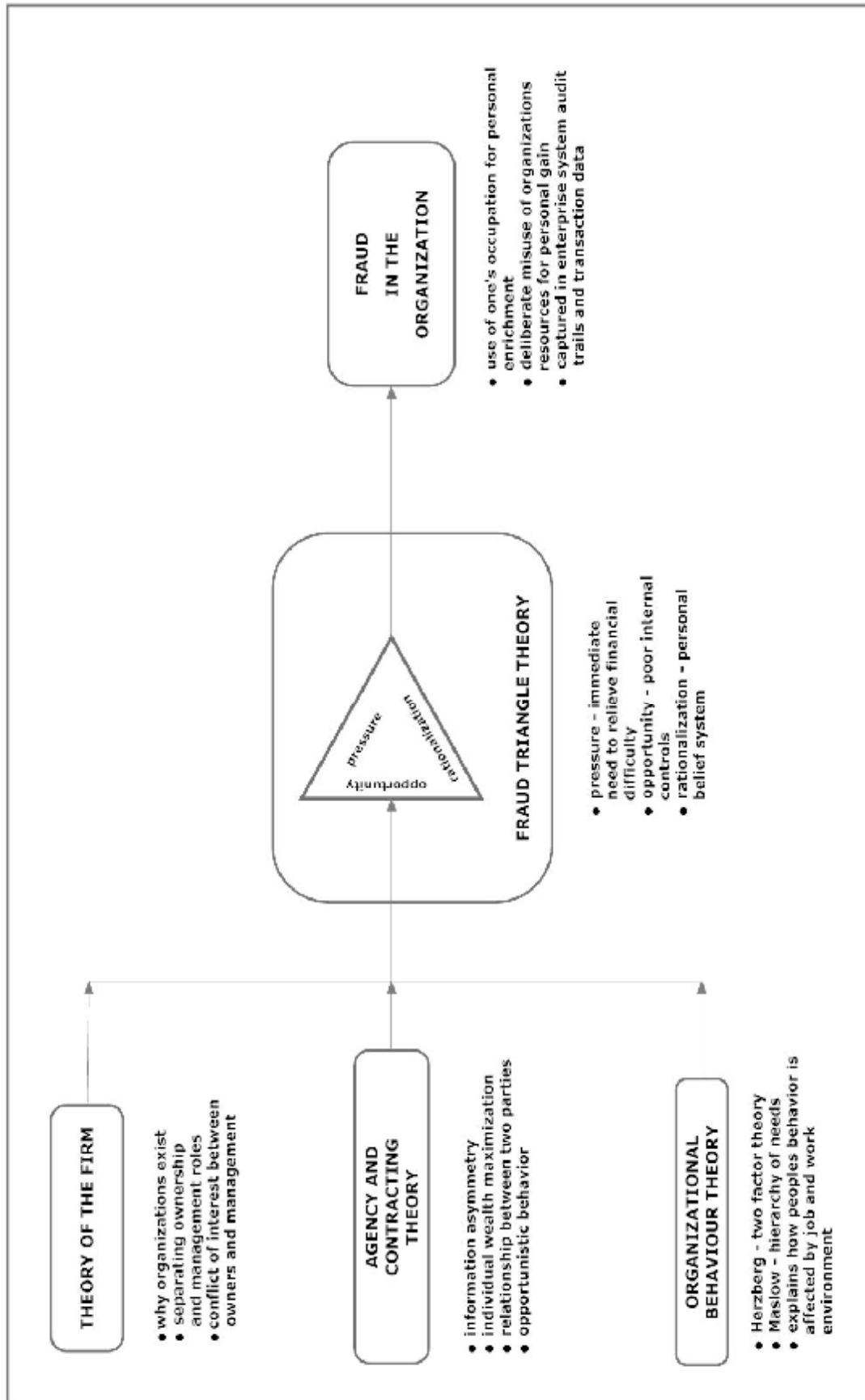


Figure 2.6: Theoretical foundation for research

discussed i.e. organisational behaviour theory, theory of the firm, agency and contracting theory, purely inform the research in providing evidence for opportunities to commit fraud based on the fraud triangle theory. The concept of opportunity is the main factor of the fraud triangle theory that provides a basis for this research as this concept and its antecedent characteristics are identifiable in a digital environment. These characteristics can be used to proactively detect potential fraud in an enterprise system by analysis of its transaction data and audit trails.

Perpetrating a fraud requires more than just an awareness of the types of fraud that can occur in an organisation. Even though all of the conditions of the fraud triangle may support the occurrence of a fraud, explicit knowledge of how to perpetrate fraud is required. Understanding the way fraudsters think after taking the decision to perpetrate a fraud is essential in the design and development of an efficient and effective model for proactive detection of potential fraud.

2.4. Framework for perpetrating fraud

Every organisation is unique. The existence, configuration and proper implementation of internal controls vary greatly among organisations and therefore no assumptions can be made about any of these characteristics. Some organisations may have strong controls in place, others may have weak controls and others yet may have none (ACFE 2010). The challenge in developing a proactive fraud detection model is the ability to adapt to the specific organisational situation and to effectively identify high probability fraudulent activities.

The ACFE Fraud Tree (Figure 2.1) is a helpful tool to understand the various types of frauds that occur commonly in an organisation. However, it does not inform investigators of how frauds occur in practice and how they can be prevented or detected in an organisation. Several cases of fraud are available in the literature (Appendix 1) yet it is rare to find a case that mentions exactly how the fraud was perpetrated, what business systems were used, and the part they played in the perpetration of the fraud. Proactive tests for detecting potential fraud in an organisation are also well-documented in the literature (Coenen 2008 ; Singleton et al. 2008 ; Wells 2008 ; O'Gara 2004 ; Potla 2003). However there appears to be a gap in the area that examines fraud from the perspective of a fraudster, starting with the desire to steal assets and ending with actions that are typically performed in an enterprise system to perpetrate fraud.

The fraud triangle theory identifies three key elements that need to be present for a fraud to occur i.e. pressure, opportunity and rationalisation. Presuming that the elements of pressure and rationalisation pre-exist, a fraudster will seek opportunities to perpetrate fraud (Murphy and Dacin 2011 ; Cressey 1950). During this process a fraudster experiences a series of thoughts related to the imminent fraudulent activity. He may enact several fraud scenarios mentally until a suitable one is found. Firstly, a fraudster will ascertain what type of asset to steal i.e. services, goods or cash. Once the type of asset is identified the next predicament is how to steal it. Up to this point a fraudster has made a conscious decision to steal the asset and has identified a general scheme to commit the theft (Figure 2.7).

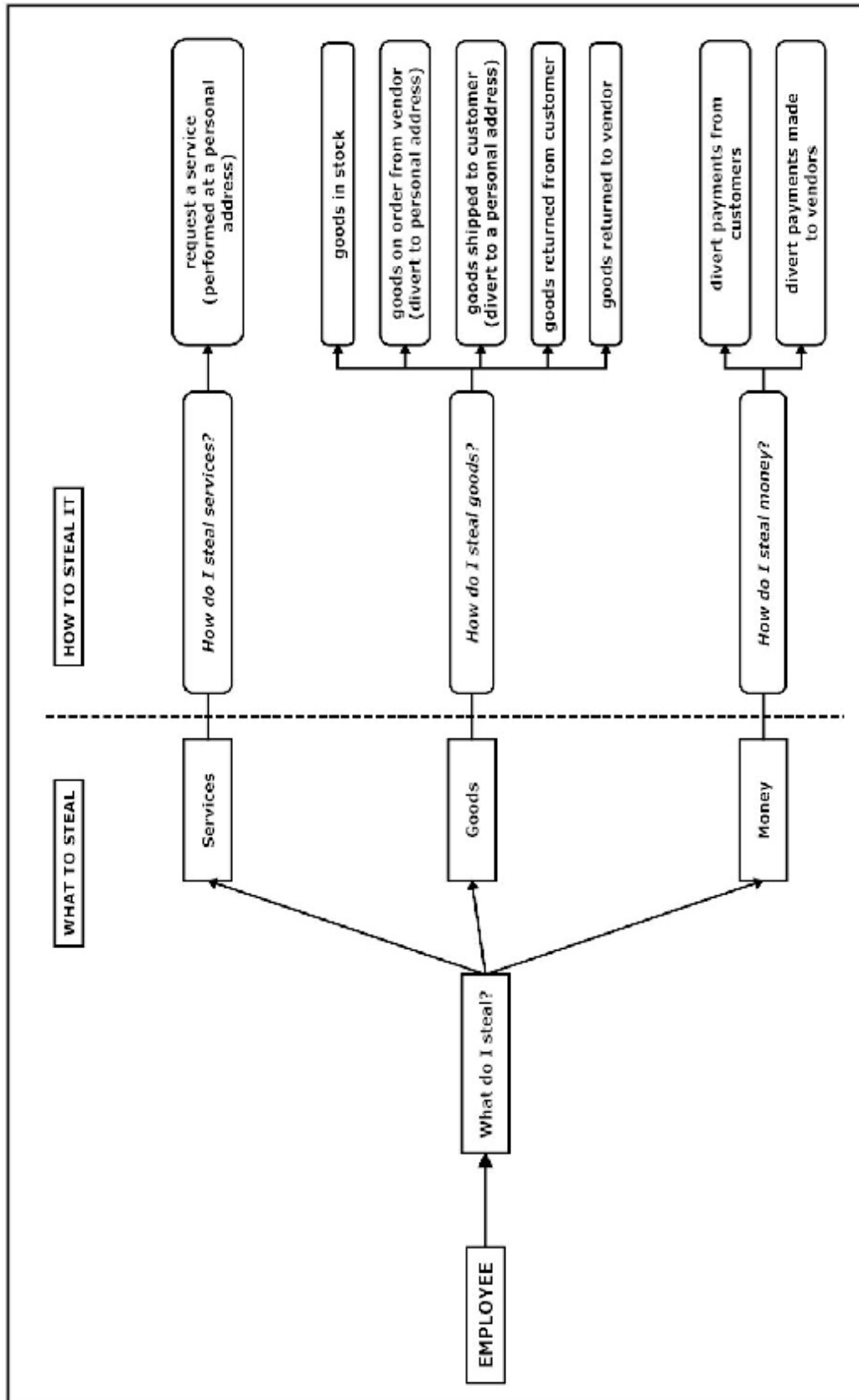


Figure 2.7: Fraud perception model (FPM)

Source: adapted from (Wells 2011 ; ACFE 2010 ; Albrecht et al. 2009 ; Best et al. 2009)

The next step is to establish a specific method of perpetrating the fraud. The method may entail a series of steps to be taken to achieve the desired outcome of committing the fraud, and concealing it to avoid detection. An example of a billing fraud is the theft of money (Figure 2.8) that an organisation intends to pay to a vendor namely **Employee/What do I steal/Money/How do I steal money/divert payments made to vendors**. Another example is the theft of goods on order by an organisation namely **Employee/What do I steal/Goods/How do I steal goods/Goods on order from vendors (divert to personal address)**.

The perpetration of billing fraud requires the creation of a shell company and the submission of fictitious invoices to an organisation for payment (Best et al. 2009 ; O'Gara 2004 ; Greene 2003b ; Wells 2002a ; Bologna 1992). To successfully perpetrate a billing fraud a fraudster must be able to:

- i). create or modify vendor master records; and
- ii). enter an invoice for payment.

(Padhi 2010 ; Best et al. 2009 ; SAP-AG 2009 ; Narayan 2008)

Vendor master records can be created or modified in the following ways.

- i). Create a fake vendor – a fake vendor is added to the list of vendors in the vendor master file. This vendor is not in a legitimate business relationship with an organisation.
- ii). Temporarily modify an existing vendor (flipping) – details of an existing legitimate vendor are temporarily changed, a payment is processed, and details are changed back to their original settings.

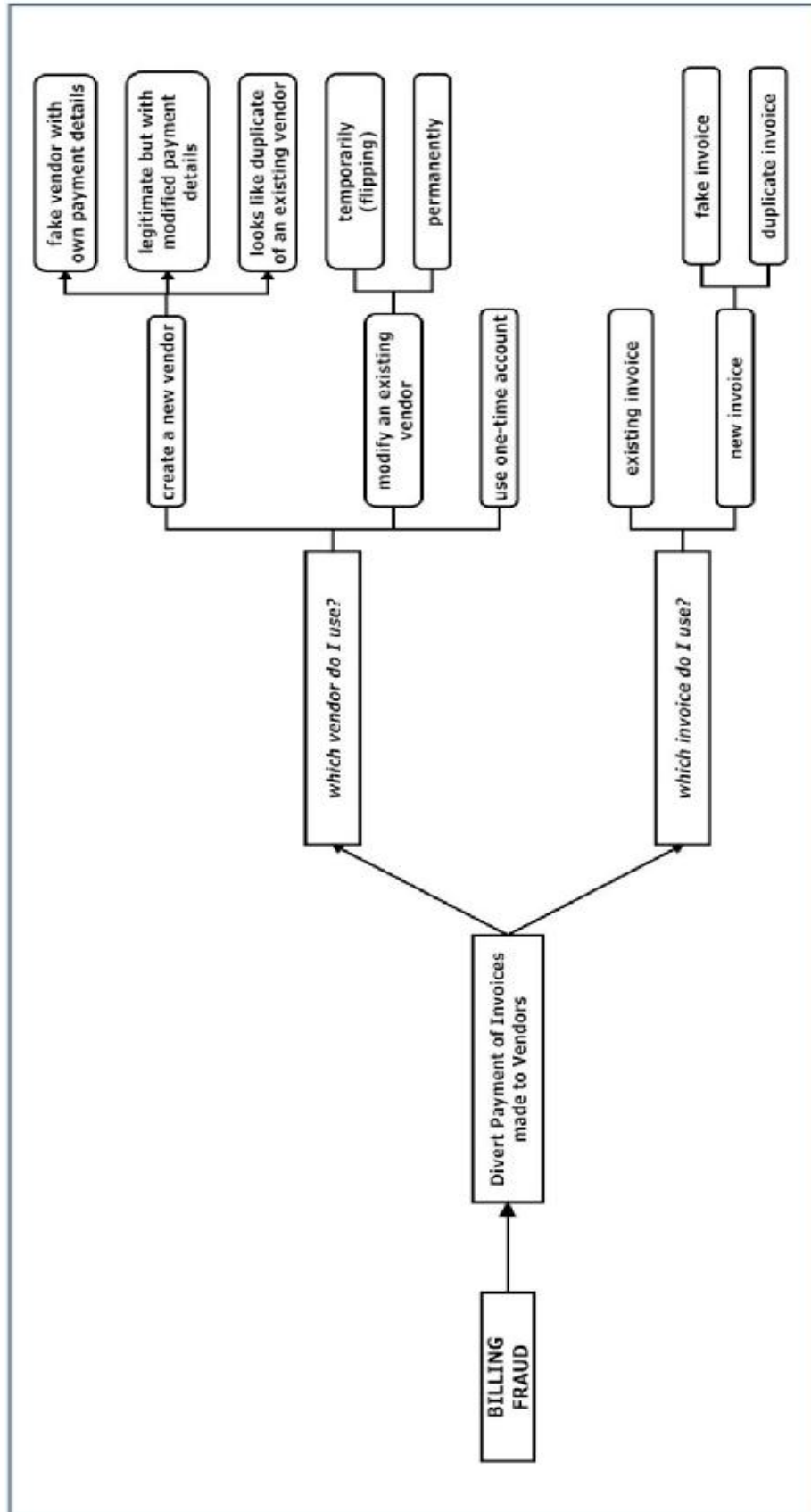


Figure 2.8: High-level fraud scenarios model (HFSM)

Source: adapted from (Wells 2011 ; ACFE 2010 ; Albrecht et al. 2009 ; Best et al. 2009)

- iii). Permanently modify an existing vendor - details of an existing legitimate vendor are permanently changed. This method is suited to vendors that no longer conduct business with an organisation but their details still exist in a vendor master file.
- iv). Use of one-time account – this account is used for vendors that an organisation rarely deals with. No vendor specific data is stored in an account; it is entered during document entry.

(Best 2008 ; Singleton et al. 2008 ; O'Gara 2004 ; Wells 2002a)

Invoices may be submitted for processing by:

- i). creating a fake invoice;
- ii). using a legitimate invoice; or
- iii). creating or using a duplicate invoice.

(Best et al. 2009 ; Singleton et al. 2008 ; Best 2005 ; O'Gara 2004)

In order for a fraud to occur all three conditions of the fraud triangle must be present. Once this situation exists a fraudster mentally envisages and plans a fraud as shown in the Fraud Perception Model (FPM) as a framework for perpetration of billing scheme frauds. (Figure 2.7). The portion of the FPM dealing with vendor fraud is extended into the High-level Fraud Scenarios Model (HFSM) (Figure 2.8). Together the FPM and HFSM form the framework for perpetrating vendor frauds in organisations. Fraud symptoms need to be identified, as recognising and cataloguing these symptoms are essential in the development of a model for proactive detection of potential fraud.

2.5. Fraud scenarios

Various known fraud scenarios or 'footprints' are used in the design of a prototype fraud detection tool in Chapter 4. In the HFSM developed in the previous section (Figure 2.8) it is necessary for the fraudster to decide on two issues i.e. "*which vendor do I use*" and "*which invoice do I use*". These two decisions are integrated and the resulting 3X4 Fraud Matrix (FM) (Table 2.5) represents known possibilities for compromising vendors and entering compromised invoices into a system. Several real cases of these fraud symptoms are provided in Appendix 1. In the following paragraphs each of the symptoms of the matrix is examined in more detail.

Scenario 1: Fake invoice for a legitimate vendor

A perpetrator creates a fake invoice. The fake invoice resembles an invoice from a legitimate vendor. Payment details in the vendor master data are temporarily changed to the perpetrators. The invoice is processed for payment. Once payment is made through an organisation's payment run the vendor's payment details are then changed back to their original values. This changing of bank account details is referred to as 'flipping'.

Scenario 2: Fake invoice for a dormant vendor

A perpetrator searches through the vendor master data to identify vendors that are inactive or dormant. These vendors may be dormant for several reasons e.g. an organisation no longer requires their products or services, they may have gone out of business or they may have merged with another business. Payment details in the vendor master data are changed to a perpetrator's. A fake invoice is created and processed for payment.

Vendor				
	Temporarily change (flipping)	Permanently change	One-time account	New and fake
Fake invoice	Change vendor payment details temporarily. Pay invoice. Change payment details back to original.	Create fake invoice for existing vendor. Change payment details permanently as vendor no longer conducts business with organisation. Pay invoice	Create fake invoice for existing non-accomplice vendor. Make payment to self using the one-time account.	Create fake vendor (shell). Create fake invoice for payment to fake vendor.
Legitimate invoice	Change vendor payment details temporarily. Pay existing invoice to self. Subsequent invoices may be paid to the vendor.	Permanently change vendor payment details. Pay invoice to self.	Modify payment details when paying invoice using one-time account.	<i>It is not possible to have a legitimate invoice in this group</i>
Duplicate invoice	Pay one invoice to vendor. Change vendor payment details temporarily. Pay duplicate invoice to self. Change payment details back to original.	Pay one invoice to vendor. Modify vendor payment details permanently and pay duplicate invoice to self.	Enter invoice twice, once for payment to vendor and once for payment to self.	Make copy of legitimate invoice. Pay vendor. Use copy to pay fake vendor with payment details modified.

Invoices

Table 2.5: Fraud matrix (FM)

Source: adapted from (Wells 2011 ; ACFE 2010 ; Albrecht et al. 2009 ; Best et al. 2009)

Scenario 3: Fake one-time invoice

A perpetrator creates a fake invoice. The fake invoice resembles an invoice from a legitimate vendor but with the perpetrator's payment details. A real non-accomplice vendor or fake vendor may be used. The invoice is processed for payment using the one-time account.

Scenario 4: Fake invoice for a fake vendor

A perpetrator creates a master record for a fake vendor with his/her payment details. Fake invoices are created for this vendor and submitted for processing. These invoices are paid during regular payment runs.

Scenario 5: Legitimate invoice

A perpetrator uses an invoice from a legitimate vendor. Payment details in the vendor master data are temporarily changed to the perpetrators. The invoice is processed for payment. Once the payment is made through an organisation's payment run the vendor's payment details are then changed back to their original values. When the vendor complains about not receiving payment a duplicate invoice is requested and it is processed for payment using the vendor's usual payment details.

Scenario 6: Legitimate one-time invoice

A perpetrator uses an invoice from a legitimate vendor that an organisation deals with infrequently. These invoices are processed through the one-time account as no vendor master data exists. Invoices are submitted for processing using the perpetrator's payment details. When the vendor complains about not receiving

payment a duplicate invoice is requested and it is processed for payment using the vendor's payment details.

Scenario 7: Duplicate invoice for a legitimate vendor

An invoice arrives from a legitimate vendor. The invoice is processed for payment. Payment is made through an organisation's payment run. A perpetrator then enters a duplicate invoice in the system. Payment details are temporarily modified in the vendor master data with the perpetrator's details. The duplicate invoice is processed for payment and a perpetrator receives the payment. Once the payment is made through an organisation's payment run the vendor's payment details are then changed back to their original values.

Scenario 8: Accidental duplicate invoice for a legitimate vendor

A duplicate invoice is accidentally entered into the system and payments are scheduled for different payment runs. A perpetrator takes advantage of this duplicate invoice. Before payment is made for the duplicate invoice (i.e. before the 2nd payment run) the perpetrator temporarily modifies payment details in the vendor master data by replacing these details with his own. Once the second payment run is made the vendor's payment details are then changed back to their original values.

Scenario 9: Duplicate invoice for a fake vendor

A perpetrator creates a master record for a fake vendor with his payment details. The perpetrator then duplicates an invoice from a legitimate vendor. The duplicate

invoice resembles the legitimate invoice but has the perpetrators payment details. The invoice is processed and payments are made during the regular payment run.

The perpetration of billing fraud requires creation of a shell company and submission of fictitious invoices for payment. The nine scenarios describe various ways fraudsters' may successfully perpetrate billing fraud by creation or modification of vendor master records, and entering fake invoices for payment. These scenarios are the basis for strategies for proactive detection of potential fraud.

2.6. Proactive fraud detection

Proactive detection of potential fraud requires continuous monitoring of an organisation's transaction data. Continuous monitoring increases the possibility of detecting fraudulent activities (Coderre and Warner 1999 ; Potla 2003). The traditional or manual audit approach is limited because it reviews only a small percentage of a large population of transactions. Large accounting data files with several thousands of transactions are difficult to analyse or monitor manually in real-time. The alternative therefore is to automate this process by using the power of information technology (Kotb and Roberts 2011 ; Broady and Roland 2008 ; Singleton and Singleton 2007).

Good internal controls require that no single employee be given too much responsibility over business transactions as this may place them in a position to commit and conceal fraud. Segregation of duties is achieved when the following functions are separated (Romney and Steinbart 2009):

- i). authorisation – approving transactions;

- ii). recording – data entry; and
- iii). custody – receiving and processing payments.

Separation of vendor maintenance, invoice entry and payment can therefore significantly reduce the risk of billing frauds in the absence of collusion among employees (Best et al. 2009 ; Little and Best 2003 ; Srinidhi 1994). However, weak, incomplete or a lack of segregation of duties often provides opportunities for fraud to be perpetrated (ACFE 2010 ; KPMG 2010 ; KPMG 2009). Early detection of fraud can limit losses and prevent the recurrence of such activities. Furthermore the Sarbanes-Oxley Act (SOX) has significantly increased corporate organisations responsibility for prevention and detection of financial fraud (Best et al. 2009 ; ITGI 2006). Therefore business executives are searching for improved ways to detect fraud (Tackett 2007) by using information technology. The essential steps in detecting fraudulent activities are (Figure 2.9):

- i). understanding the business or operations;
- ii). performing a risk analysis to identify the types of frauds that can occur;
- iii). cataloguing the symptoms that the most likely frauds would generate;
- iv). using computer technology to identify fraud symptoms;
- v). analysing the results; and
- vi). investigating suspicious transactions.

(Albrecht et al. 2009)

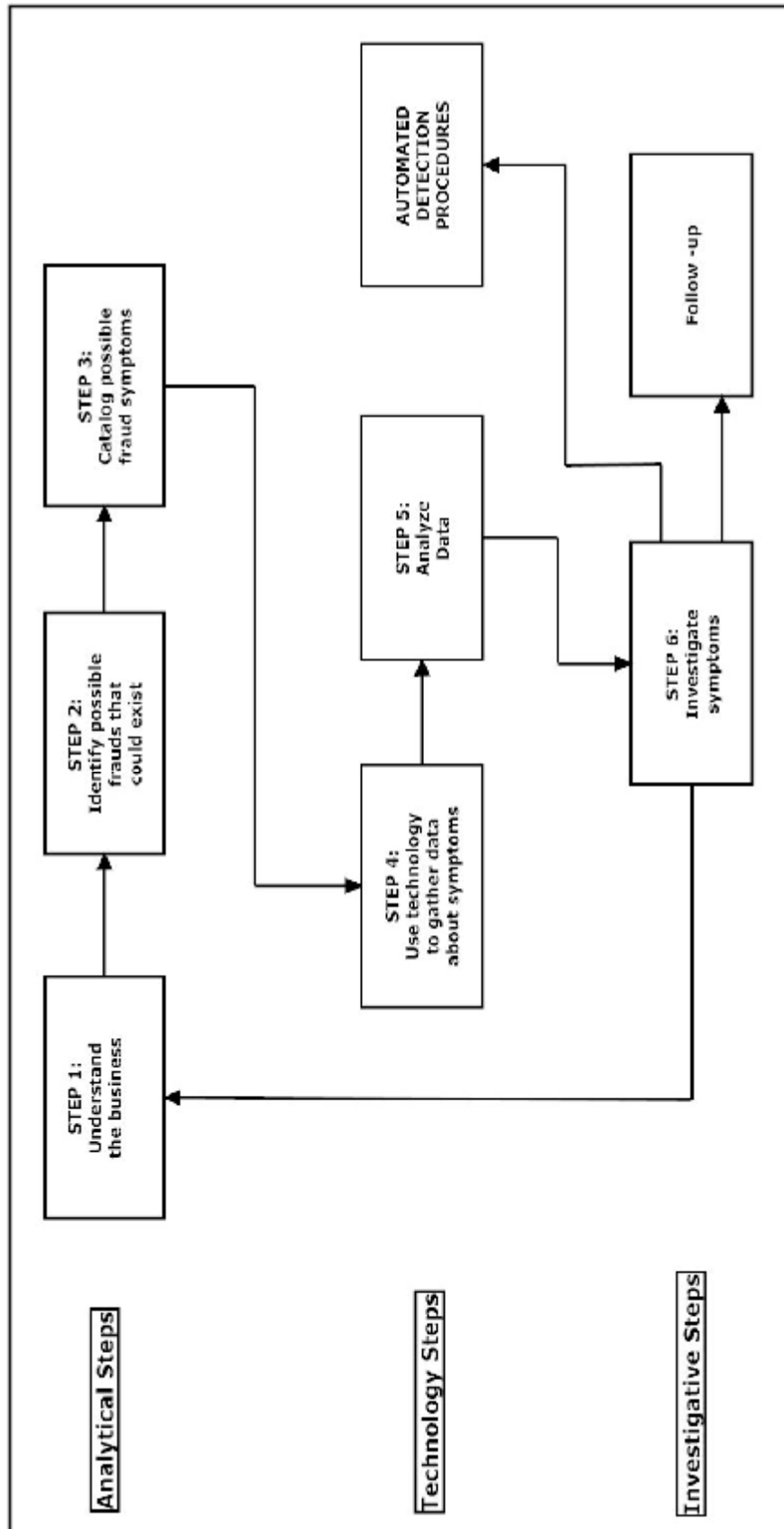


Figure 2.9: Fraud detection process

A catalogue of fraud symptoms is developed in Chapter 4. This catalogue forms the basis of fraud detection strategies used in this study. Audit trail data is analysed for symptoms of potential fraud using computer technology. Activities identified as potentially fraudulent require further investigation by an internal auditor to determine whether actual fraud exists. Using computer technology to analyse audit trail data facilitates identification of potentially fraudulent activities from many thousands of transactions.

Early identification and reporting of suspicious activities to an auditor may potentially reduce the negative impact on an organisation in the form of loss of revenue, goodwill or reputation (Coderre and Warner 1999).

Automated systems that continuously monitor for key fraud symptoms can be a major deterrent of fraud (Best et al. 2009 ; Potla 2003 ; Coderre and Warner 1999). By analysing data and searching for specific patterns or combination of activities, potentially fraudulent activities can be identified shortly after they occur. Data analytics can be used to detect suspicious activities that have already occurred as well as to proactively determine the propensity for frauds occurring in the future (Edge and Falcone Sampaio 2009).

Technology based continuous fraud monitoring and detection is an important area for improvement in the fight to reduce fraud in organisations. Using data analytics to proactively detect potential fraud is an important priority for many organisations. These organisations, however, do not have the expertise and therefore require assistance in using technology to identify and detect fraud (KPMG 2010 ; KPMG

2009 ; KPMG 2008). Presently only 2.6% of organisations are using data monitoring to proactively detect fraud (ACFE 2010) (Figure 2.10) while 53% of organisations report fraudulent activities (KPMG 2010). A fraud detection system that is innovative, intuitive and simple to use may possibly contribute to an improvement in these statistics.

The fraud landscape is dynamic, fast-moving and ever changing. Fraudsters are becoming more sophisticated in their use of technology and in their ability to commit and conceal fraudulent activities. As a result, fraud detection techniques must continue to evolve (Gill 2009). A key element of any effective proactive fraud detection system is the continuous monitoring of an organisation's transaction data. This increases the possibility of detecting past and future fraudulent activities if appropriate strategies for continuous monitoring of an organisation's transaction data are put in place.

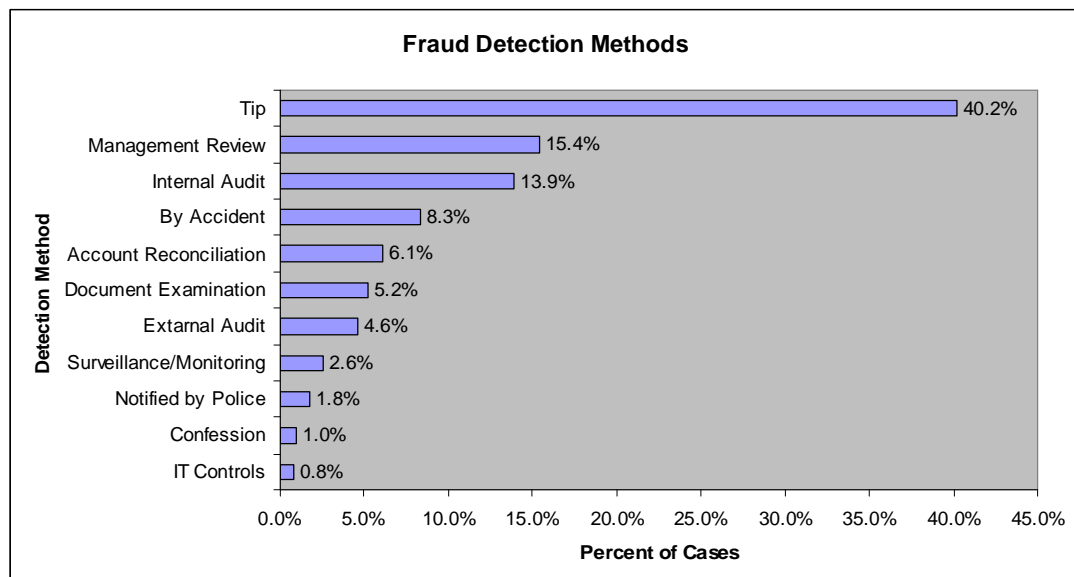


Figure 2.10: Detection of occupational fraud

Source: ACFE (2010)

2.7. Continuous monitoring strategies

Examining the transaction process from transaction entry through to the posting in the general ledger has traditionally been conducted manually, or with Computer Assisted Audit Techniques (CAATs) (Flowerday and von Solms 2005) on a retrospective and cyclical basis, usually many months after business activities have transpired. This is a massive task as it may involve millions of transactions hence only a sample is examined. Even when using CAATs, transactions are examined in batches and generally only sampled. If the transactions are examined in their entirety then it is usually done retrospectively.

Increasing use of information technology has made it necessary for auditors to perform audits using software tools that run on personal computer systems (Kotb and Roberts 2011). Software tools automate standard audit processes and procedures by using productivity software such as Microsoft Office, and specialised data analyses software such as ACL or IDEA. Using CAATs to conduct audits is termed 'auditing through' the computer (Vasarhelyi et al. 2004) as opposed to 'auditing around' the computer, which treats the computer as a 'black box' (Matthews 2006). CAATs can be very powerful tools that enable auditors to perform various routine audit tasks. However, several difficulties exist that impede successful CAAT implementations (Arens et al. 2007). Among these are: i) sufficient depth of IT knowledge is required to develop complex data extractions and programming tasks; ii) sufficient time and skill is required to develop and test a CAAT; iii) development and setup cost of CAAT may be prohibitive; iv) access to specialised training by audit staff is required in order to create complex CAAT reports; v) audit staff may not have the aptitude or interest in developing CAAT competencies; and vi) CAAT resources may not be

available to audit teams when required. Additionally, CAATs are limited because they do not take advantage of new technologies to automate and integrate audit processes and procedures, and they do not provide sufficient response to the new challenges of auditing in the modern global digital economy (Vasarhelyi et al. 2004).

The modern global digital economy has significantly changed the way business is conducted and therefore the traditional approach to auditing can no longer be of real value to business performance or regulatory compliance (Coderre 2005 ; Rezaee et al. 2002). Most organisations are conducting their business activities online and in real-time. This necessitates continuous monitoring and auditing which enables internal auditors to perform their analyses of key business systems in real- or near real-time (Kuhn Jr and Sutton 2010 ; Alles et al. 2006 ; Coderre 2005 ; Alles et al. 2002 ; Kogan et al. 1999). Therefore, by necessity, the audit process has evolved from the traditional manual approach to one that is computer-based.

Continuous auditing was devised in the famous Bell Labs in 1989 as a way to provide constant monitoring of AT&T's billing system (Rutgers 2010). Continuous monitoring, also referred to as 'continuous auditing', is the analysis of transaction data in a real- or near real-time basis against a set of predetermined rule sets (Kuhn Jr and Sutton 2010). It enables auditors to provide "*some degree of assurance on continuous information simultaneously with, or shortly after disclosure of information*" (Rezaee et al. 2002 p.150). It is a step in the path of the evolution of the financial audit from manual to computer-based methods. Continuous monitoring is only feasible if it is implemented as: i) a fully automated process; and ii) a process with instant access to relevant events (Kogan et al. 1999). The widespread adoption

of computer-based accounting information systems in general, and enterprise systems in particular, has contributed to the increasing demand for continuous monitoring (Vasarhelyi et al. 2004).

The basic need for continuous monitoring has increased in the new global digital economy as organisations become more complex and demand more integrated business processes. Many types of management and control information needs exist and in the real-time economy these can only be satisfied by continuous monitoring and assurance. Collapses of multi-national organisations have imposed strict regulatory and legislative requirements on organisations. The Sarbanes-Oxley Act (SOX) of 2002 has highlighted fraud and its detection, with the emphasis being on improving internal controls to reduce the risk of financial fraud (Best et al. 2009). As the number of technology-enabled businesses continue to grow, new needs arise for continuous monitoring and assurance concerning: i) changes in the environment and industry; ii) the existence and effectiveness of controls; iii) increased human resource risks; iv) increased use of outsourced processes; v) process continuity and integrity; and vi) coherence between endogenous and exogenous factors (Vasarhelyi et al. 2004).

Providing continuous monitoring and assurance in the new technology-enabled business environment require a comprehensive understanding of the way businesses organise their activities (Alles et al. 2006). While organisations have always had to measure and monitor their activities, paper-based systems (i.e. accounting journals and ledgers) relied on pre-filtered and aggregated measures which were typically recorded after significant time had elapsed. Modern computer-based systems make it

possible to measure and monitor business processes at an unprecedented level of detail in a real- or near real-time basis.

Large scale implementations of enterprise systems have resulted in many organisations being highly automated and fully integrated. The development of this enterprise system environment provides the necessary infrastructure for the effective evolution of the assurance function from a periodic event to an ongoing process through the use of continuous monitoring applications (Goode and Lacey 2011 ; Debreceeny and Gray 2010 ; Kuhn Jr and Sutton 2010). Two major approaches to continuous monitoring and auditing exist. These are Embedded Audit Modules (EAMs), and Monitoring and Control Layer (MCL).

2.7.1. Embedded Audit Modules (EAMs)

EAMs are Computer Assisted Audit Techniques (CAATs) that enable the continuous monitoring of enterprise systems (Debreceeny et al. 2005). EAMs are software modules that are built into application programs and are specifically designed to continuously capture and monitor audit related data (Groomer and Murthy 1989). If a pre-programmed constraint is violated an alert is produced, an auditor is informed, and transaction data are saved in a file (Best et al. 2009 ; Debreceeny et al. 2005 ; Weber 1999 ; Groomer and Murthy 1989)

Weber (1999) defines EAMs as software modules that are placed at specific points within an enterprise system to gather data about transactions or events that auditors deem material. EAMs are therefore intended to detect and capture data as

transactions are processed in an enterprise system. When a violation occurs the offending transaction can either be rejected, or it can be allowed and the error is logged. Enterprise systems are designed to process transactions efficiently and promptly. It is therefore not practical to disallow every offending transaction from being processed. Depending on the severity of a violation, some transactions could be conditionally processed whilst others are rejected. The level of severity of errors that would cause a transaction to be rejected needs to be negotiated and accepted by the client organisation (Groomer and Murthy 1989).

In the context of integration with enterprise systems EAMs should have the following characteristics (Debreceeny et al. 2005):

- i). an end-user environment that allows the auditor to establish a set of queries to test transaction integrity constraints either from a pre-defined suite of queries, the modification of the attributes of pre-defined queries, or by the creation of new queries by the construction of simple scripts;
- ii). a process for registration (embedding) and scheduling of these queries;
- iii). a method for running these queries against the flow of transactions for violations either continuously or temporally
- iv). a capacity for reporting violations electronically (e.g. by email); and
- v). an ability to copy the transaction details of the violations to secondary storage

EAMs have the potential to capture data about all transaction errors and violations. Consequently they can be used as compliance or substantive testing tools because of their capability to continuously monitor transaction data for errors or violations. If EAMs are used throughout an accounting period, they will record data about the operation of controls (compliance-testing), as well as data about actual transaction errors (substantive-testing). Thus they can facilitate dual-purpose testing and constitute a comprehensive auditing tool (Groomer and Murthy 1989).

EAMs and other CAATs are an important step in the advancement of continuous monitoring and auditing (Alles et al. 2002 ; Rezaee et al. 2002 ; Kogan et al. 1999 ; Vasarhelyi and Halper 1991). When the objective of a continuous audit is clearly defined then EAMs can support the process by appropriate monitoring of an enterprise system. The requirement of the Sarbanes-Oxley Act (SOX) for improved prevention and detection of financial fraud (Best et al. 2009 ; ITGI 2006), has executives searching for improved ways to detect fraud (Tackett 2007) by employing information technology, including EAMs and other CAATs, to monitor internal controls (Ernst and Young 2002 ; PricewaterhouseCoopers 2002).

Regarding this approach, its benefits, drawbacks, technologies and processes are extensively discussed in the literature (Alles et al. 2006 ; Debreceeny et al. 2005 ; Alles et al. 2004 ; Groomer and Murthy 2003 ; Groomer and Murthy 1989). There is limited evidence of adoption of EAMs despite several factors that indicate the possibility of wide-spread adoption (Debreceeny et al. 2005 ; Groomer and Murthy 1989). Vasarhelyi and Halper (1991) provided early evidence of the feasibility of

continuous monitoring and auditing. Debreceeny et al. (2005) demonstrated the use of EAMs in the high-volume transaction flows of a telecommunications enterprise.

Research seems to indicate that the EAM approach runs into several difficulties (Kuhn and Sutton 2006 ; Debreceeny et al. 2005). Vasarhelyi and Halper (1991) expressed several challenges to their development and implementation, including issues related to design and utilisation of system resources. Since EAMs are software applications they require computer processing time to execute. This imposes an overhead on a system which in turn negatively impacts monitoring processes. Although this overhead can be overcome by adding additional hardware and software resources, these additional investments have costs associated with them. There is also the concern about having 'foreign' software embedded within an organisation's enterprise system, and this software being the responsibility of a third party (Best et al. 2009 ; Alles et al. 2006 ; Debreceeny et al. 2005). The maintenance of EAMs can also be difficult given the changes, updates and modifications that routinely take place in enterprise systems. There is also legal liability issues should an EAM damage a host enterprise system in some way, a liability that external auditors may be keen to avoid. Implementation of EAMs in the SAP enterprise system requires the writing of an Advanced Business Application Programming (ABAP) script at the application layer. This requires auditors who wish to make use of EAMs to have expert knowledge of ABAP programming and the database structure of an organisation. Thus there appears to be considerable technical and legal barriers to the development of audit routines that operate continuously on enterprise systems. These factors have impeded the adoption of EAMs in enterprise systems (Debreceeny et al. 2005; Alles et al. 2006).

2.7.2. Monitoring and Control Layer (MCL)

The Monitoring and Control Layer (MCL) introduced by Vasarhelyi et al. (2004) is an alternative continuous monitoring and auditing approach to EAMs. The MCL approach does not replace EAMs, instead it offers an alternative solution to cater for different circumstances (Kuhn Jr and Sutton 2010). In this approach the continuous monitoring and auditing system is separate from a client's enterprise system. The MCL approach is a stand-alone system that relies on comparisons of extracted transaction data with pre-determined constraints that allow for continuous monitoring of systems and identification of violations (Du and Roohani 2007).

The MCL approach uses 'middleware' to extract data from an enterprise system for external monitoring and analysis (Kuhn Jr and Sutton 2010 ; Alles et al. 2008 ; Du and Roohani 2007 ; Alles et al. 2006 ; Kuhn and Sutton 2006 ; Murthy and Groomer 2004 ; Vasarhelyi and Halper 1991). The main elements of the MCL architecture are (Vasarhelyi et al. 2004 p.13): *"a data capture layer; a data filtering layer; relational storage; measurement standards layer; an inference engine; an analytic layer; alarms and alerting layer; and a reporting platform."*

The MCL primarily operates as a discrepancy-based audit monitoring tool i.e. audit by exception (Vasarhelyi et al. 2004). The MCL continuously captures enterprise data and analyses it to detect any deviations from the norm. Whenever a significant exception is detected, an alarm is produced and sent to pre-determined compliance personnel by using relevant delivery technologies such as emails, telephone calls or pagers. When an alarm is delivered, compliance personnel will need to review the

evidence in order to identify the underlying problem. Any further investigations are at the discretion of internal auditors.

The continuous monitoring system that makes up the MCL (i.e. workstations, operating systems, database and application software) resides outside a client's network and is controlled by an auditor. The system receives periodic data updates from the client enterprise system, (i.e. not in real-time), that is processed inside an application. The system monitors key operational analytics, compares them with pre-defined standards and creates exception reports for any potential problems. Any violations that trigger automatic alerts to an auditor are stored inside an application and not inside a client's enterprise system.

Systems developed using the MCL approach are intended to match a client's base enterprise system. This improves efficiency and effectiveness of communication between systems for the purpose of data extraction and continuous monitoring. Whilst this approach provides client-independence, it does impose a significant overhead on auditing organisations as they need to acquire significant IT resources to develop, maintain, store, and continuously monitor each client's enterprise system (Kuhn Jr and Sutton 2010). Since an organisation and the external environment are constantly changing a continuous monitoring MCL system has to be constantly updated and improved. Tests and analytics performed by a MCL have to be re-examined and modified to incorporate the constantly changing environment in which firms operate. These changes may range from minor updates of fraud patterns to major changes in MCL's structure requiring a complete redesign of a MCL (Vasarhelyi et al. 2004).

The MCL approach has a long history in the context of accounting information systems, dating as far back as 1991 (Kuhn Jr and Sutton 2010 ; Alles et al. 2008 ; Du and Roohani 2007 ; Alles et al. 2006 ; Kuhn and Sutton 2006 ; Murthy and Groomer 2004 ; Vasarhelyi and Halper 1991). The MCL unites various IT systems in an organisation into one integrated platform that allows for seamless real-time data exchange. In organisations using enterprise systems, a MCL would be part of auditing systems thereby providing an auditor with the capability to drill down to individual transactions and then to roll up data for analysis at any level of aggregation. Direct 'read-only' access to enterprise systems data helps maintain independence of audits, ensures that data cannot be altered and that data integrity is preserved. It is this capability that facilitates real-time identification and confirmation of potentially fraudulent activities in an enterprise system.

In summary, the two major approaches to continuous monitoring and auditing are Embedded Audit Modules (EAMs) and Monitoring and Control Layer (MCL). Monitoring activities conducted in both EAM and the MCL approaches focus on transaction data, which is monitored for violations of pre-set standards or unusual patterns. EAMs have several critical deterrents to adoption including issues relating to system design and maintenance, client independence and legal liability. MCLs are separate stand-alone external systems that operate independently of the information system to be monitored but are linked into a system. They rely on comparisons of extracted transaction data with pre-determined constraints to identify violations. This separate design has profound implications for the design of a general model for continuous monitoring and auditing as it eliminates any conflict between a MCL and an enterprise system. The MCL approach is therefore a major facilitator for

implementing continuous monitoring and auditing in enterprise systems. It is also the approach used in this research.

2.8. Enterprise Systems

Enterprise systems comprise a significant share of the market (Rothenberger et al. 2010). In 2010, vendors of larger enterprise systems (i.e. SAP, Oracle and Microsoft) shared 53% of the ERP market (Panorama 2011).

Enterprise Systems (ES, sometimes also known as enterprise resource planning or ERP systems) are integrated or packaged business software systems that are designed to streamline the flow of data in an organisation with the intention of matching the physical flow of goods from raw materials to finished products. This flow of data may extend well beyond the boundaries of an organisation to include the supply chain at one end, and the customer at the other (Kamhawi 2008 ; Koch and Wailgum 2008 ; Presley 2006 ; Norris et al. 2000). Enterprise systems adopt a structured approach to optimising an organisation's internal value chain by linking various components of an enterprise through sharing of common data. For example, when a sale is recorded, this information is used to update other areas in an enterprise such as inventory, procurement, invoicing and recording of all related ledger postings (Deshmukh 2006 ; Musaji 2002 ; Norris et al. 2000). Enterprise system, therefore have several distinctive characteristics (Norris et al. 1998).

- i). Multi-functional in scope – it tracks financial results (dollars), procurement (material), sales (people and goods) and manufacturing (people and resources).

- ii). Integrated in nature, that is, when a piece of data is entered regarding one function, data regarding other functions is changed.
- iii). Modular in structure, that is, it can be used in a way that is as expansive or narrow as required.

There are five main reasons why an organisation may consider implementing an enterprise system (Koch and Wailgum 2008 ; Ehie and Madsen 2005 ; Xu et al. 2002).

- i). Integration of financial information - many different versions of information may exist e.g. finance has its own set of revenue numbers, sales has another version, and different business units may each have their own version of how much they contributed to revenues. ESs create a single version of data across an enterprise because everyone uses the same system. Hence, integrity of information is preserved.
- ii). Integration of customer order data—ESs contain a common repository for customer orders from the time a customer service representative receives it until the loading dock ships merchandise and finance sends an invoice. By having this data in a common integrated database system, rather than across many disconnected systems, organisations can keep track of orders more easily, and coordinate manufacturing, inventory and shipping among many different locations at the same time.

- iii). Standardisation of manufacturing processes – ESs provide standard methods for automating many of the steps of a manufacturing process thereby saving time and improving employee productivity.
- iv). Reduction in inventory – by making the manufacturing process flow more efficient, ESs can help users' better plan deliveries to customers, thereby reducing finished goods inventory at warehouses.
- v). Standardisation of human resources (HR) data – in organisations that span multiple business units, HR may not have a unified method for managing employees' time and communicating with them about benefits and services. These issues are readily resolved by using ESs.

Essentially enterprise systems are therefore organisation-wide information systems that integrate all aspects of a business. Enterprise systems use an underlying database system to store business data. All business data associated with a system are stored in one database. This ensures that data is consistently shared across an organisation.

Enterprise systems software are available from several vendors including SAP, Oracle and Microsoft, collectively having 53% of ES installations worldwide (Panorama 2011). However for several years, Germany-based enterprise software company SAP has consistently been the market leader (SAP 2010 ; Lager and Tsai 2008). In 2010 Gartner (Gartner 2010) recognised SAP as the leading vendor of

enterprise systems software, accounting for 22.4% of market share. For this reason SAP is used in this project.

2.9. SAP Enterprise System

Since 1972 SAP-AG has been providing business software solutions to the market, starting with SAP R/2, SAP R/3, and the evolution towards mySAP. SAP defines mySAP as a complete e-business platform that provides a range of solutions for its customers and users. mySAP is therefore the common name that SAP uses for all technologies that it produces. It is an open, flexible and comprehensive business solution that integrates both SAP and non-SAP applications. mySAP is capable of integrating internal business processes as well as providing a collaborative platform among business partners (Hernandez 2002).

SAP belongs to the family of enterprise systems or enterprise resource planning (ERP) systems. Organisations may develop an enterprise system by acquiring and integrating specialised sub-systems (e.g. financials and payroll) with internally developed ones (e.g. equipment hiring), thereby producing a 'multi-vendor' solution. SAP is a 'single-vendor', packaged enterprise system, but it has the ability to integrate with non-SAP systems (Best 2005).

The SAP enterprise system provides the following functionality (Vogel and Kimbell 2005).

- i). Cross-functional scenarios that match daily business processes.
- ii). Analytics that provide information when and where it is required by employees.

- iii). Dashboards with simple interfaces that place everything an employee needs to get their job done in one place.
- iv). Embedded roles that make it easy to provide the information and functions each person needs based on their role in an organisation.
- v). Industry solutions that extend and customise generic ESs to make sense in every industry setting.
- vi). Real-time integration through SAP NetWeaver. SAP NetWeaver provides a platform to integrate all internal organisational systems as well as integrating with third-party systems. It also acts as a platform on which applications can be built.

Traditional SAP applications are categorised in three core functional areas: financial, human resources, and logistics. SAP also develops and provides special modules that complement core modules. These are targeted at vertical industries such as retail, manufacturing and government. These packages are known as SAP Solutions for Industries (SAP-AG 2009 ; Hernandez et al. 2006 ; Vogel and Kimbell 2005). There is a special set of modules, known as cross-application (CA) modules, which is positioned between technical and functional areas of the system and cover things such as the business workflow, CAD integration, and documents handling. The core areas include hundreds of business processes to address all the needs of modern business applications. There are many modules within these areas that can work equally well as stand-alone products. For instance, there are organisations that might decide to use only certain modules of the SAP Core application suite such as

accounting, sales and distribution, and manufacturing. Customising is required for all modules.

SAP financial modules provide an overall picture of accounting functions, with extensive report facilities to allow for fast decision-making support. The financial area contains the following module groups: i) FI - Financial accounting, ii) CO – Controlling, iii) EC - Enterprise controlling, iv) IM - Capital investment management, and v) TR – Treasury. As new versions of the product are released new modules are added (Padhi 2010 ; Hernandez et al. 2006 ; Vogel and Kimbell 2005).

The Human Resources (HR) module includes all of the necessary business processes required to efficiently manage all the needs of an organisation's human resource management. Data is entered once in the HR module and is made available to other related applications, namely accounting, plant maintenance, or business workflow. The HR module provides support for salary administration and payroll, work schedule models, planning, travel expenses, and so on (SAP-AG 2009 ; Hernandez et al. 2006 ; Vogel and Kimbell 2005).

The Logistics module manages all processes involved in a supply chain, from procurement of raw materials to final delivery and billing of a customer. These modules contain comprehensive business processes and several tools for decision support. Modules integrate seamlessly with almost every other SAP module, from financial and controlling modules to human resources (Padhi 2010 ; Hernandez et al. 2006 ; Vogel and Kimbell 2005).

Many organisations have realised that SAP solutions are important to their success. SAP solutions provide an organisation with competitive advantage. Several Fortune 500 companies use SAP exclusively for their core day to day operations (Gartner 2010 ; BOS 2009), which include accounting and financial applications, procurement, order processing and supplier management, inventory management and HR management and payroll functions. SAP enterprise systems are fully integrated, enabling transactions to be processed organisation-wide, and consequently they contribute to an overall improvement in an organisation's operational efficiency (Wailgum 2008).

Enterprise systems generate millions of transactions annually. While most of these are legal and routine transactions, a small number may be fraudulent. The enormous amount of generated transactions makes it difficult to find these few instances among legitimate transactions. An auditor may extract a small sample of these during a financial audit. Many fraudsters rely on this to conceal fraud. The problem becomes overwhelming and is growing worse. Many organisations are considering using data analytics and information technology (IT) to detect fraud. Using IT to proactively detect potential fraud enables organisations to monitor and analyse large transaction datasets in real- or near real- time, a task that cannot practically be accomplished by an internal auditor. The prototype software being developed in this study will exploit SAP audit trails for proactive detection of potential fraud.

2.10. Audit trails

Audit trails are records of users' activities within an information system (Best 2005 ; NIST 2005 ; Gopalakrishna 2000). Audit trails are maintained by operating systems,

database systems and enterprise systems (Best et al. 2004). Data captured in an audit trail is dependent on what events are being audited by a system (SAP-AG 2009). In conjunction with appropriate tools and procedures, audit trails can assist in detecting potentially fraudulent activities and anomalous user behaviour.

Audit trails provide a means to accomplish several security related objectives including the following (Best et al. 2009 ; Best 2005 ; NIST 2005).

- i). Review of access: Audit trails allow examination of histories of access by individual users or groups of users, showing actions performed or attempted. Audit trails also can report which users have performed specific functions, such as changes to vendor master records or entry of vendor invoices. Analysis of audit trails may also reveal limitations in an organisation's security model and its implementation.
- ii). Review of changes in security: Changes made to the security of the system can be reviewed periodically by an independent person for authorisation and integrity.
- iii). Review of attempts to by-pass security: Audit trails may be reviewed for attempts and repeated attempts by users and intruders to perform unauthorised functions.
- iv). Deterrent against attempts to by-pass security: Users should be aware of the existence of audit trails and their routine review as a deterrent against attempts to by-pass security.
- v). Fraud detection: Audit trails can be used to detect potential fraud by searching for 'red flags'. Fraudulent activity may be

perpetrated by real users acting in their own name, by users acting in collusion with other users, by real users masquerading as other users, or by intruders masquerading as authorised users. In each case, actions of these 'users' are recorded in audit trails and these can be scrutinised for activities that are recognised as 'red flags' for particular types of fraud.

A system can maintain several different audit trails concurrently. There are typically two types of audit records: i) keystroke monitoring; and ii) event-based logs. Keystroke monitoring records keystrokes entered by a computer user and a computer's response during an interactive session. Event-based audit logs contain records describing system events, application events, or user events. An audit trail should include sufficient data to establish what events occurred and who caused them (Broady and Roland 2008 ; NIST 2005).

System audit records are used to monitor and fine-tune system performance. Application audit records may be used to detect flaws in applications, or violations of security policy within an application. Users' audit records are used to hold individuals accountable for their actions. Users' audit trails monitor and log users' activities in a system or application by recording events initiated by users. An analysis of users' audit records may expose a variety of security violations, which may range from simple browsing to detection of attempts to defraud an organisation (NIST 2005 ; Gopalakrishna 2000). For the purpose of this research the term 'audit trail' will hereinafter refer to 'user audit record.'

Audit trails attempt to capture a chronological list of steps that are necessary to start a transaction through to its completion. Audit trails can range from being very simplistic to extremely complicated. The complexity depends on the number of steps involved in a transaction. For example, an audit trail on payment of a vendor invoice begins with a receipt of an invoice. The invoice is tracked through accounts payable, all the way through to payment in order to settle a debt (Tatum 2010).

The purpose of auditing is to verify that financial information correctly reflects the economic condition of an organisation. Audit trails provide an auditor with a detailed account of users' activities in an enterprise system and can therefore be an effective tool in managing financial resources of an organisation. An auditor reviews transaction data to verify its validity, completeness and correctness. There are three types of audit trails that record activities at different levels of detail, namely system audit trails, application audit trails and user audit trails. The focus of this research is on user audit trails in ESs, as they monitor and record user activity in a system or application by recording events initiated by users.

2.11. Enterprise system audit trails support for fraud detection

"...the best planned and implemented audit trail is of limited value without timely review of the logged data."

(NIST 2005 p.219)

Understanding the structure of audit trails and data they capture is a key factor in designing systems for proactive detection of potential fraud. Denning (1987)

introduced the concept of using audit trails to detect anomalous user behaviour.

Denning's model consists of six main components.

- i). *Subjects*: Initiators of activity on a target system – normally users.
- ii). *Objects*: Resources managed by the system-files, commands, devices, etc.
- iii). *Audit records*: Generated by the target system in response to actions performed or attempted by subjects on objects-user login, command execution, file access, etc.
- iv). *Profiles*: Structures that characterise the behaviour of subjects with respect to objects in terms of statistical metrics and models of observed activity. Profiles are automatically generated and initialised from templates.
- v). *Anomaly records*: Generated when abnormal behaviour is detected.
- vi). *Activity rules*: Actions taken when some condition is satisfied, which update profiles, detect abnormal behaviour, relate anomalies to suspected intrusions, and produce reports.

Denning's model is rule-based and exploits audit trails to search for and report abnormal user behaviour. The basic objective of the model is to monitor audit records looking for deviations in usage. Audit records should at minimum include the following data:

- i). *Action*: Operation performed by the subject on or with the object, e.g., login, logout, read, execute.
- ii). *Exception-Condition*: Denotes which, if any, exception condition is raised on the return. This should be the actual exception condition raised by the system, not just the apparent exception condition returned to the subject.
- iii). *Resource-Usage*: List of quantitative elements, where each element gives the amount used of some resource, e.g., number of lines or pages printed, number of records read or written, CPU time or I/O units used, session elapsed time.
- iv). *Time-stamp*: Unique time/date stamp identifying when the action took place.

Each audit record specifies a subject (initiator of the action in the target system) and an object (receptors of actions, i.e. programs, files, databases, and so on). Each activity is observed without regard for authorisation, as the assumption is that access controls in the system permitted the action to occur. The target system is responsible for maintaining audit records.

Denning also investigated several methods for developing activity profiles. These profiles characterise the behaviour of a given subject with respect to a given object. This serves as a signature to describe normal activity and a means to detect anomalous activity. When a new audit record matches a pattern in an activity profile an audit-record rule checks for any abnormal behaviour. If such a behaviour is detected an anomaly report is generated.

Audit trails may be reviewed: i) periodically; ii) as needed (triggered by a security event); iii) automatically in real-time; or iv) some combination of these. Audit trails can be used to retrospectively determine or review what events have occurred. Reviewers need to know what 'red flags' to look for i.e. what is normal activity and what is suspicious activity. Audit trail review is made easier if the audit trail can be analysed by user ID, terminal ID, application name, date and time, or some other set of parameters to run reports of selected data (NIST 2005).

Enterprise systems maintain several different audit trails concurrently. **Security audit logs** record details of each user action such as successful logins, failed logins, starting a transaction, failed attempts to start transactions, automatic locking of a user's account because of multiple failed logins, creation of new roles/profiles and changes in user master records (Best et al. 2009 ; Best 2005 ; Best 2000). Configuration of the security audit log defines what events are recorded e.g. only failed activity may be recorded. Audit trails may be retained for periodic review and then archived.

Changes to master records, such as those for vendors, are an important aspect in detection of fraud. A master record must be created or modified (e.g. temporarily changing a vendor's banking details) in order for a system to pay a vendor invoice for a shell company electronically or by cheque, Records of such changes in master records show user identification, type of change (e.g. create, delete, change), and contents of fields created/deleted/changed. These activities should be flagged as suspicious and selected for further review.

Accounting audit trails facilitate tracing of accounting transactions from a source to updating of accounting balances and tracing of account balances back to source transactions. They provide an organisation with the ability to maintain sufficiently detailed records to answer enquiries from customers or vendors, to produce detailed reports and monthly statements. Master record changes and accounting audit trails are retained on-line usually for the entire fiscal year, and archived for several years to satisfy requirements of taxation and company legislation.

Fraud research in enterprise systems has primarily focussed on fraud prevention rather than detection (Goode and Lacey 2011 ; Albrecht et al. 2009 ; Coderre 2005 ; Best 2005). Several vendors of enterprise systems software provide Governance, Risk, and Compliance (GRC) software tools. These tools are prevention oriented and are intended to ensure good internal controls and to automate creation and management of these controls (Broady and Roland 2008 ; Hernandez et al. 2006). This research aims to make use of various audit trails available in SAP enterprise systems, namely security audit logs, records of changes to master records and accounting audit trails, to develop innovative methods of detecting and reporting fraudulent activities to audit and compliance personnel.

2.12. Gaps in the literature

Given the pervasiveness of enterprise systems (Kuhn Jr and Sutton 2010 ; Rothenberger et al. 2010 ; Singleton and Singleton 2007), there are some noticeable gaps in the literature regarding fraud detection in ESs. Additional research is necessary to advance awareness, relevance, and practicality of continuous fraud

detection in enterprise systems (Kotb and Roberts 2011). The study being conducted in this research will make a contribution to the literature by addressing gaps identified in the literature review.

There appears to be limited research in continuous monitoring and fraud detection (Du and Roohani 2007 ; Kuhn Jr and Sutton 2010). Debreceeny et. al. (2005) noted that at the time of their study there was limited research to support the use of Embedded Audit Modules within enterprise systems. Since their study there has been limited research projects (Alles et al. 2006 ; Kuhn and Sutton 2006) that examined continuous monitoring and auditing in enterprise systems. Furthermore, there appears to be limited research in developing a generalised model for proactive detection of potential fraud in enterprise systems (Goode and Lacey 2011). This gap is addressed by the primary research question.

Can a generalised model for proactive detection of potential fraud in enterprise systems be developed?

Prior research on continuous auditing does not appear to deliver a model that allows an audit to be carried out proactively and continuously without difficulties (Hunton et al. 2004). Additional study is required to develop approaches of continuous auditing that are specifically applicable to auditing of financial transactions in enterprise systems (Debreceeny and Gray 2010). This gap is addressed by research sub-question 1 and its research propositions.

SQ1: How do enterprise systems support proactive detection of potential fraud in financial transactions?

Research is required in the development of innovative approaches to continuous fraud detection in organisations that use enterprise systems, and to demonstrate how this can be done efficiently and effectively (Du and Roohani 2007 ; Rezaee et al. 2002). An issue often raised in the literature relates to information overload from alerts when implementing continuous fraud detection systems (Alles et al. 2008 ; Alles et al. 2006 ; Kuhn and Sutton 2006). A related issue deals with integrity of the data used for continuous fraud detection (Kuhn Jr and Sutton 2010). Another issue evident from fraud surveys is the average time taken to detect fraud appears to be increasing (KPMG 2010 ; KPMG 2008). These are important issues as there appears to be a potential demand for efficient and effective implementation of continuous detection of potential fraud in organisations (Daigle and Lampe 2004). This gap is addressed by research sub-question 2 and its research propositions.

SQ2: How can detection of potential fraud in enterprise systems be effectively and efficiently automated to ensure minimal auditor interaction?

2.13. Conclusion

Fraud is inherent in all organisations. It is a multi-billion dollar industry that is continuing to grow annually. Fraud costs Australia approximately \$3 billion annually, and its frequency and financial impact continues to grow. Many organisations are poorly prepared to prevent and detect fraud. Fraud detection strategies are intended to complement fraud prevention strategies. They are expected

to quickly and efficiently identify those frauds that have circumvented preventative measures so that an organisation can take appropriate corrective action.

There are many factors that motivate individuals to commit fraud. Several theories that contribute to the perpetration of fraud were identified in the literature. A common theme in each of the theories was that conflict of interest between business owners and its employees may possibly lead to fraudulent behaviour when an opportunity presented itself. Perpetrating a fraud required more than just an awareness of the types of fraud. Explicit knowledge of how to perpetrate a fraud is also required. Understanding how fraudsters think is essential in the design and development of an efficient and effective model for proactive detection of potential fraud.

Proactive detection of potential fraud requires continuous monitoring of an organisation's transaction data. Continuous monitoring increases the possibility of detecting fraudulent activities. The traditional or manual audit approach is limited because it reviews only a small percentage of a large population of transactions. Large accounting data files with several thousands of transactions are difficult to analyse or monitor manually in real-time. The alternative therefore is to automate this process by using information technology.

The basic need for continuous monitoring has increased in the new global digital economy as organisations become more complex and demand more integrated business processes. The fraud landscape is dynamic, fast-moving and ever changing.

Furthermore, fraudsters are becoming more sophisticated in their use of technology and in their ability to commit and conceal fraudulent activities.

Large scale implementations of enterprise systems have resulted in many organisations being highly automated and fully integrated. The development of this enterprise system environment provides the necessary infrastructure for effective evolution of the auditing function from a periodic event to an ongoing process through use of continuous monitoring applications.

The focus of fraud research in enterprise systems has principally been on fraud prevention rather than detection. The focus of this research is therefore, to explore and develop innovative methods for proactive detection of potential fraud in enterprise systems by continuous monitoring and analysis of its audit trails.

CHAPTER 3

Research Design and Methodology

3.0. Introduction

A review of the literature on fraud and its detection in enterprise systems is undertaken in Chapter 2. The primary research question and sub-questions are developed from gaps identified in the literature and posted in this Chapter.

A scientific methodology should form the basis of any academic research project which claims to add value to the body of knowledge. A scientific methodology is a systematic observation of nature (Chadwick et al. 1984). Scientific understanding proceeds by way of constructing and analysing models of segments or aspects of reality being studied. The purpose of these models is not to give a mirror image of reality, nor to include all its elements in their exact sizes and proportions, but rather to single out and make available for intensive investigation those elements which are decisive (Baran and Sweezy 1970).

A methodology may serve as a set of rules for reasoning whereby evaluation of facts can be used to draw inferences. The use of a methodology infers some competence in logical reasoning (Remenyi 1990). A researcher may be able to establish or verify some theories and these must be validated by some form of empirical evidence.

This Chapter discusses the research design and methodology which are adopted by this study.

3.1. Study design

Research design ensures that the evidence collected enables the researcher to answer the research questions in an unambiguous way (De Vaus 2001). Collection of evidence requires a researcher to specify the type of evidence needed to answer the research questions. Research design refers to the structure of an enquiry and therefore it is a logical problem and not a logistical one (Yin 2002). It is not related to any particular data collection method nor any particular type of data (De Vaus 2001) and can use qualitative or quantitative data. It is inaccurate to equate a particular research design with either qualitative or quantitative data as research design can use any type of data collection method.

When designing a research project a researcher will bring their particular assumptions and beliefs to the research that will influence the approach used in the study (Patton 1990). This influence is termed a research paradigm. A paradigm is a model or set of values and beliefs that gives direction to the researcher (Creswell 2005). There are three (3) paradigms: positivism; interpretive and critical (Smith 2003). Table 3.1 provides a summary of these three paradigms and their key characteristics.

A positivist perspective assumes that the method for gaining information and knowledge should be independent of a researcher, have certainty through data that measures reality composed of discrete elements and be replicable as research is

Table 3.1: Research paradigms

Source: (Smith 2011 p.5)

Positivist	Interpretive	Critical
1. What is the approach modelled on?		
Classical investigation founded in the physical sciences.	Historical, literary and existential studies in which subjective understandings of subjects are significant	Marxist and interpretive studies which focus on the insights and judgements of the subjects.
2. What does it assume about reality?		
Reality is unitary and can only be understood by empirical and analytic methods.	There are multiple realities which require multiple methods for understanding them.	There are multiple realities which are problematic through distorted communication.
3. What is the foundation of data?		
Disciplined rules for observation.	Meanings are the basis of data: meaning precedes logic and fact.	Means are found in language and social behaviours and they precede logic and fact.
4. How is observation done?		
Through and unambiguous rules which are not modified by setting and are totally independent of it.	Through the social, linguistic and cognitive skills of the researcher.	Interpretive methods, plus critical self-reflection concerning the grounds of observation.
5. What is generated?		
Evidence and generalisable laws which are not affected by contexts and have nothing to do with the way they were discovered. Objectivity depends on removal of error and bias.	Knowledge which is dependent on the process of discovery. The integrity of the findings depends upon the quality of the social, linguistic and cognitive skills of the researcher in the production of data analyses and conclusions.	Knowledge which falls within the interpretive framework, but which also serves the purposes of assisting personal liberation and understanding, and emancipation from forces constraining the rational independence of individuals.
6. What interests are inherent?		
Prediction and control, technically exploitable knowledge, and explanation.	Understanding at the level of ordinary language and action. Discovering the meanings and beliefs underlying the actions of others.	Interpretive interests and those which underlie other forms of inquiry. Radically improving human existence. Practical and public involvement in knowledge formation and use.
7. What values are inherent?		
Science and scientific knowledge are inherently value-neutral.	Science and scientific knowledge have both to be interpreted in terms of values they represent.	Science and knowledge are never value-neutral: they always represent interests.

limited to natural, physical and material approaches. Data collection and analysis is structured, with the researcher not intervening in the phenomenon of interest (Perry et al. 1999).

An interpretive perspective assumes that human actions occur as a result of external influences, having both intentions and reflections. In order to develop an understanding of a phenomenon, a researcher may be required to become an active participant in it. It is important to understand the process of discovery in order to remove any ambiguity that results from a researcher's active participation in it (Smith 2011).

A critical perspective expands on the scope of an interpretive one. It focuses on ownership of knowledge and the associated social, economic and political implications. Research enquiries are usually long-term ethnographic processes and structures. Assumptions are subjective and knowledge is grounded in social and historical routines and is value-dependent and not value-free (Perry et al. 1999).

In this study, the focus is on using technology to demonstrate the *feasibility of implementing proactive detection of potential fraud in enterprise systems*. It requires a systematic, impartial and responsible approach. The positivist paradigm provides the required discipline for conducting this study. The study is a classical investigation of physical phenomena that can be understood by empirical and analytic methods. Rules for observation of data are unambiguous and not affected by the environment. The evidence obtained is generalisable and is not affected by contexts or the way it was discovered. There are two major processes of reasoning, 'deductive' (theory to observation) and 'inductive' (observation to theory). Inductive

reasoning begins with specific observations (data) from which theories may be generated or generalisable patterns may emerge. Deductive reasoning starts with theory and proceeds to predictions that follow from its application. Predictions can be verified from its application (Smith 2011). Furthermore, fraud theorist Robertson (2000) proposes that fraud examinations begin with assumptions, based on available data (facts). Assumptions are then tested to determine whether they can be proven or not. Therefore, remaining within a positivist paradigm, this study proposes a research design that incorporates laboratory experiments and field research.

Laboratory experiments are designed to measure the effects of experiments precisely and under controlled circumstances. These may range from timing aspects of a prototype system to measuring the effects of some new product on an intended target audience. Field research, being less controlled, is often conducted to observe a phenomenon in its natural environment (Oliver 1999). Most field research is cross-sectional and access of any kind may be opportunistic and subsequently denied. Researchers may therefore use a small set of case-data in support of more general, theoretical assumptions. However, the ability to make broad generalisations from a single study is necessarily limited. When evaluating validity, alternative views of the same phenomenon should be offered through a process of 'triangulation'. Combining different results from interviews, surveys, archival data collection, and so on, using both quantitative and qualitative approaches is termed 'between-methods' triangulation (Smith 2011).

Before deciding on what data to collect for a study, and how to collect it, it is necessary to decide whether to use quantitative or qualitative research methods, or a combination of both. Qualitative research design is commonly used when there is little

known about a problem or when a detailed understanding is required of a specific phenomenon. Quantitative studies generally use large samples to test numerical data by comparing or finding correlations among sample attributes so that the findings can be generalised to a population (Creswell 2005).

No one research method is more superior to another. An important determinant, however, is obtaining valid and meaningful result for a study by choosing the most appropriate method. The quantitative approach tends to collect more limited data about a large participant group, while the qualitative approach collects more rich data from a small participant group (Creswell 2005). For this study, the following outcomes are important: i) experimentation in a laboratory environment; ii) testing in the field; and iii) opinions of experts. A mixed approach is therefore suited to this study as it may provide more meaningful and valid results.

3.2. Research design

This section discusses the research design. The main research question and its sub-questions are discussed, a conceptual model is developed and propositions are formulated.

3.2.1. Research questions

This study aims to answer the following primary research question.

**Can a generalised model for proactive detection of
potential fraud in enterprise systems be developed?**

To answer the primary research question, two research sub-questions have been developed.

SQ1: How do enterprise systems support proactive detection of potential fraud in financial transactions?

This sub-question examines the possibility of detecting potentially fraudulent activities in an enterprise system. The enterprise system used for the purpose of the study is SAP. In order to answer this question the following issues need to be addressed.

- i). Can user activities be identified as potentially fraudulent?
- ii). Can user activities be monitored in an enterprise system?
- iii). What evidence is required to monitor user activities in SAP?
- iv). Can procedures be developed to detect potentially fraudulent activities in SAP?

SQ2: How can detection of potential fraud in enterprise systems be effectively and efficiently automated to ensure minimal auditor interaction?

This sub-question examines the possibility of automating detection of potential fraud in an enterprise system, as it is impractical to analyse large accounting data files manually. A MCL-based approach is stand-alone and client independent. It enables auditors to perform analyses with minimal interaction with an organisations enterprise system. In order to answer this sub-question the following issues need to be addressed.

- i). Can software be developed to automate detection of potential fraud?
- ii). Can the software be implemented on a stand-alone computer system that is separate from a SAP enterprise system?

3.2.2. Conceptual model

This research develops a conceptual model for proactive detection of potential fraud in enterprise systems. This model is derived from i) the Fraud Perception Model; ii) the High-level fraud scenarios model (HFSM); and iii) the Fraud Matrix from Sections 2.3, 2.4 and 2.5. The six essential steps in detecting fraudulent activities are discussed in Section 2.6 and Figure 2.9 (Albrecht et al. 2009). The conceptual model in Figure 3.1 incorporates the following four steps to detect fraud in enterprise systems:

- i). identify types of frauds that can occur;
- ii). catalogue fraud symptoms;
- iii). use computer technology to detect fraud symptoms; and
- iv). analyse results.

This study proposes a Monitoring and Control Layer (MCL) based model for proactive fraud detection through continuous monitoring and analysis of enterprise systems' audit trails. The proposed model has the following inherent characteristics.

- i). It is not limited to sampling a subset of an organisation's transactions, as is the case with a traditional manual audit. Therefore there is no sampling risk.
- ii). It provides frequent opportunities for identifying potential fraud. This will most likely lead to a reduction in the time taken to detect fraud, from several months to days or hours.

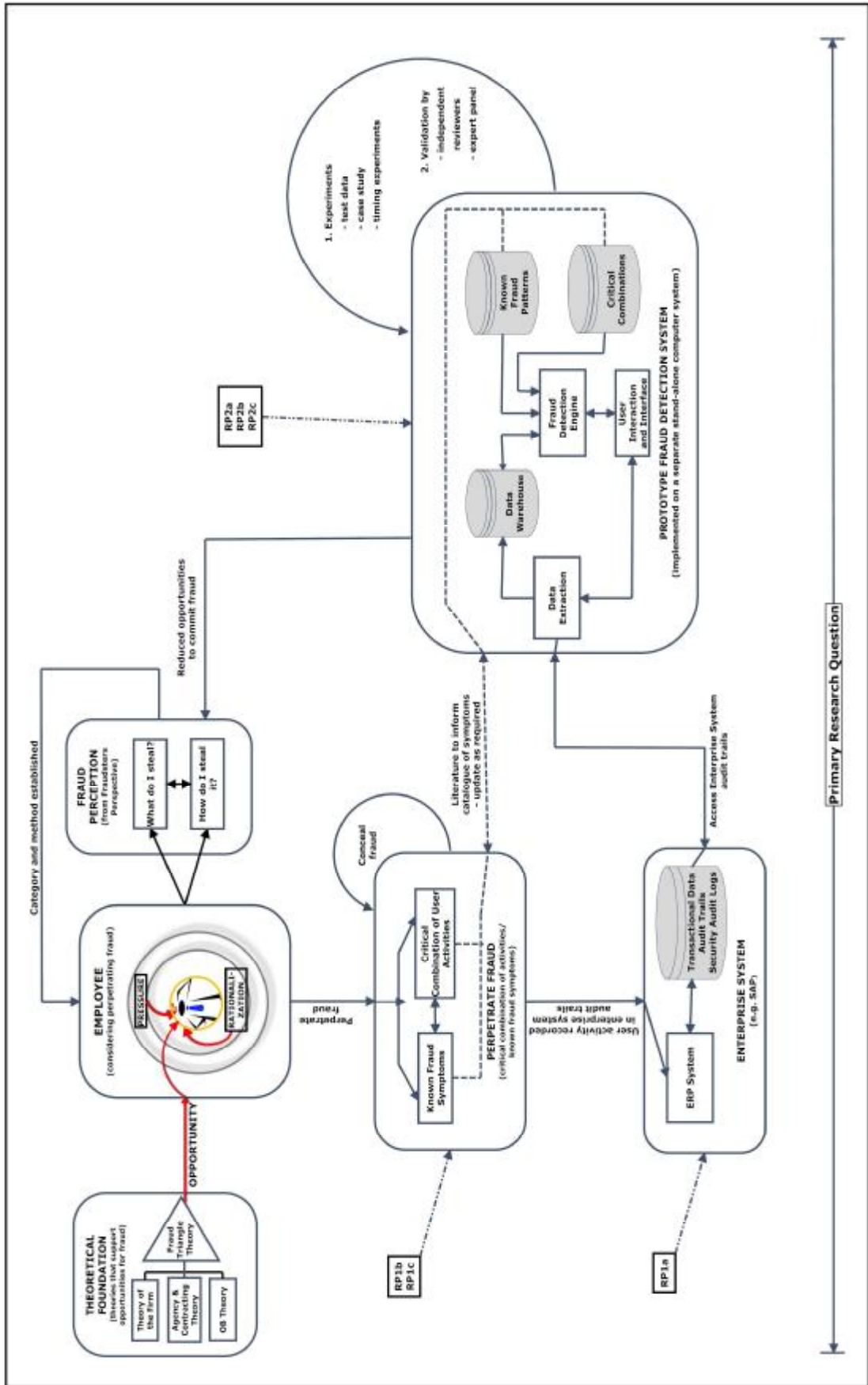


Figure 3.1: Conceptual model

- iii). In-built data analytics will assist in determining the propensity for frauds occurring in the future. This attribute may be used in identifying, and proactively correcting deficiencies in internal controls thereby leading to a possible reduction in fraud in an organisation.
- iv). Access to an enterprise system is only required for data extraction purposes. A separate system is used for continuous monitoring and data analytics. Impact on the enterprise system being monitored is therefore negligible as there is no overhead of running additional software. Independence of enterprise systems' from audit computer systems is maintained.

The conceptual model represents: i) the fundamental nature of fraud; and ii) its detection. Firstly, the model incorporates factors that motivate an individual to perpetrate fraud. It identifies mental states that fraudsters experience prior to perpetrating frauds. A fraudster may mentally enact several fraud scenarios until a suitable one is found. Once a fraudster determines '*what to steal*', the next decision is '*how to steal it*'. A fraudster has to determine a specific method of perpetrating fraud. The chosen method may entail a series of steps taken to achieve the desired outcome of; i) perpetrating a fraud and, ii) concealing it to avoid detection. The key concept identified in this part is *opportunity*. Secondly, the model focuses on detection of potential fraud in an organisation. This is achieved by:

- i). creation of a catalogue of fraud symptoms;
- ii). translation of fraud symptoms into detection strategies that can be implemented in a prototype;

- iii). design and development of a prototype; and
- iv). experimentation with enterprise system (for example, SAP) data.

The conceptual model for this research integrates components of fraud discussed in preceding sections. The model provides an understanding of the nature of fraud symptoms and its detection in enterprise systems (for example, SAP). Fraud is a complex social condition that evolves from underlying factors such as dissatisfaction or despair. The eventual outcome is that an individual is motivated to misappropriate assets that belong to an organisation.

3.2.3. Research propositions

To facilitate answering of the research sub-questions, propositions have been formulated. Each of the propositions directs attention to a specific issue that needs to be examined within the scope of the research sub-question (Figure 3.2). The propositions assist in directing the study towards the desired outcome of answering the primary research question and proving the conceptual model. The propositions are discussed below.

SQ1: How do enterprise systems support proactive detection of potential fraud in financial transactions?

To answer this research sub-question, three propositions have been formulated.

RP1a: Enterprise system audit trails document adequate data to allow retrospective monitoring of user activities.

Fundamental data required to monitor user activities in an enterprise system is established in section 3.2.3 and explained further in section 4.2. The SAP enterprise system is subsequently examined, in Chapter 4, to determine the level of support it provides for proactive detection of potential fraud.

RP1b: Violations in segregation of duties can be identified by analysing audit trails for critical combinations of user activities.

A catalogue of critical combinations of user activities, and procedures to identify user activities that violate of segregation of duties are developed in section 4.4.1.

RP1c: Potentially fraudulent transactions can be identified by investigating user activities that violate segregation of duties, match known fraud symptoms, or appear otherwise anomalous.

A catalogue of fraud symptoms and procedures to identify potentially fraudulent transactions that occur as a result of violation of segregation of duties are developed in section 4.5. The rationale is that a user performing any critical combination of user activities may be involved in perpetrating a fraud against an organisation.

SQ2: How can detection of potential fraud in enterprise systems be effectively and efficiently automated to ensure minimal auditor interaction?

To address this research sub-question, three propositions have been formulated.

RP2a: Software can be developed to identify potentially fraudulent activities and report these using an intuitive visual interface.

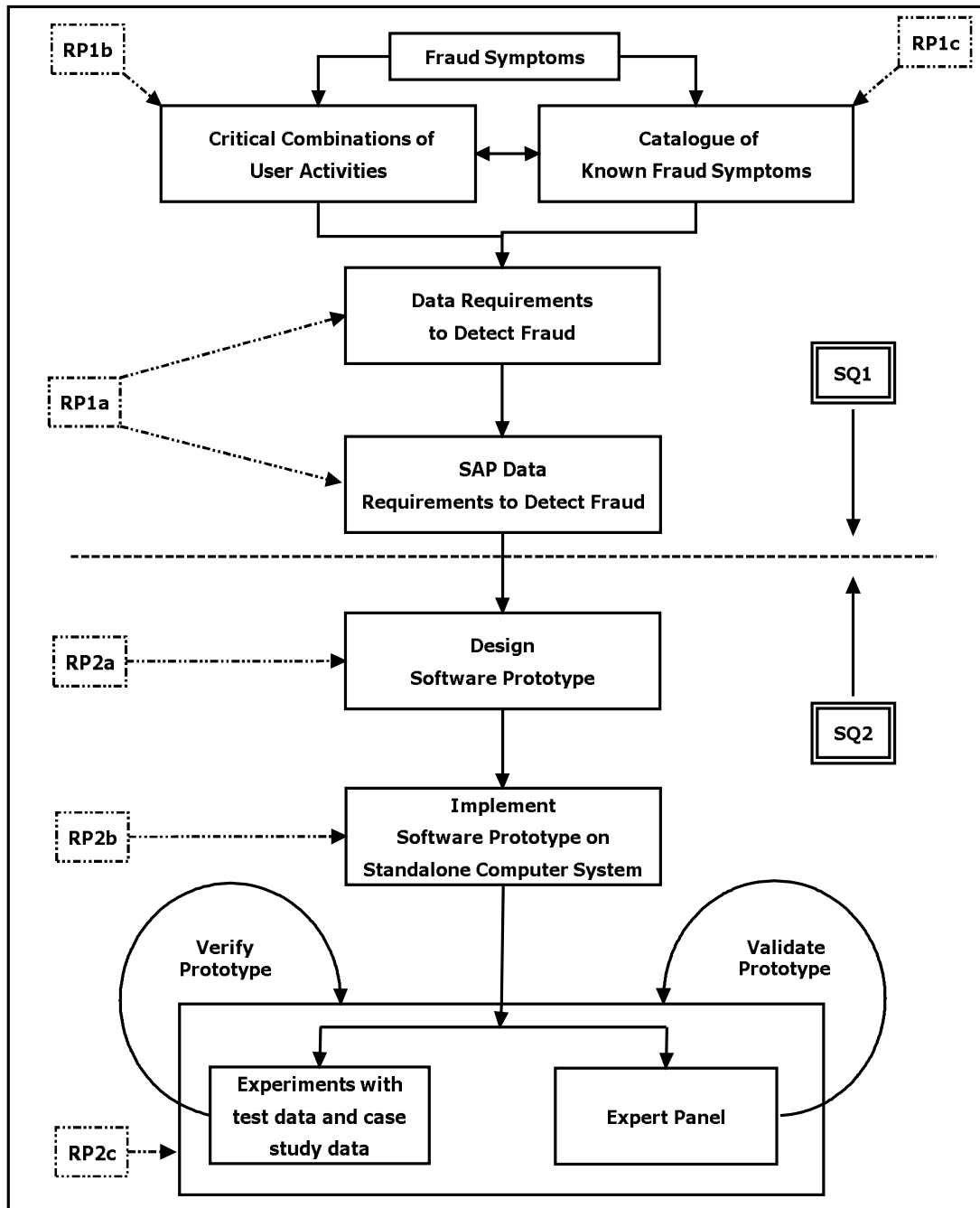


Figure 3.2: Research propositions

Prototype software for detection of potential fraud is designed in Chapter 4 and implemented in Chapter 5. Tests are performed with the prototype using simulated

test data and case study data from a SAP enterprise system. The prototype is demonstrated to independent reviewers and a panel of experts. It is refined based on the results of experimentation with test data. Reports and visualisations produced by experimenting with the prototype provide evidence in support of the feasibility of implementing proactive detection of potential fraud in practice.

RP2b: Threat monitoring and potential fraud detection can be implemented on a stand-alone external computer system operating independently of an organisation's enterprise system.

Prototype software is developed as a stand-alone application and installed on a separate computer system in Chapter 5. Experiments using test data and case study data are performed on a laptop computer, independent of an organisation's SAP enterprise system. Tests are conducted on a variety of data-sets to determine whether the prototype is able to handle real data volumes from a real organisation without difficulty. Feedback is sought from independent reviews and expert panel members to determine desirability for a proactive tool for detection potential fraud implemented on a stand-alone computer system.

RP2c: Efficiency and effectiveness of the audit process can be improved by using technology to perform continuous monitoring of an organisation's enterprise system.

Experiments using test data and case study data are performed with the prototype to provide information on processing times in Chapter 5. Results obtained from these

Table 3.1: Mapping of research questions, propositions and process

Research Question	Research Proposition	Process
<p>SQ1</p> <p>How do enterprise systems support proactive detection of potential fraud in financial transactions ?</p>	<p>RP1a</p> <p>Enterprise system audit trails document adequate information to allow retrospective monitoring of user activities.</p> <p>RP1b</p> <p>Violations in segregation of duties can be identified by analysing these audit trails for critical combinations of user activities.</p> <p>RP1c</p> <p>Potentially fraudulent transactions can be identified by investigating user activities that violate segregation of duties, match known fraud symptoms, or appear otherwise anomalous.</p>	<p>Determine data requirements to monitor user activities.</p> <p>Establish whether SAP audit trails provide the required data.</p> <p>Create catalogue of critical combinations of user activities.</p> <p>Develop procedures to identify critical combinations.</p> <p>Create catalogue of fraud symptoms to be identified.</p> <p>Develop procedures to identify fraud symptoms.</p>

...continued on next page

...Table 3.1 continued from previous page

Research Question	Research Proposition	Process
<p style="text-align: center;">SQ2</p> <p>Can detection of potential fraud in enterprise systems be effectively and efficiently automated with minimal auditor involvement?</p>	<p style="text-align: center;">RP2a</p> <p>Software can be developed to identify potentially fraudulent activities and report these using an intuitive visual interface.</p> <p style="text-align: center;">RP2b</p> <p>Threat monitoring and potential fraud detection can be implemented on a stand-alone external computer system operating independently of an organisation's enterprise system.</p> <p style="text-align: center;">RP2c</p> <p>Efficiency and effectiveness of the audit process can be improved by using technology to perform continuous monitoring of an organisation's enterprise system.</p>	<p>Develop proof of concept prototype. Use prototype to analyse SAP data. Refine prototype based on results of data analysis and feedback from focus group (expert panel).</p> <p>Develop and install software prototype on separate audit computer system. Perform tests to determine functionality.</p> <p>Perform experiments with case study data to determine processing times. Obtain feedback from independent reviewers and expert panel.</p>

tests and feedback from independent reviewers and expert panel members are used to establish support of this proposition.

Table 3.1 summarises the relationships among research questions, research propositions and the process required to answer research questions.

3.3. Research methodology

The conceptual fraud detection model developed in this study proposes that *a generalised model for proactive fraud detection in enterprise systems can be developed.*

The methodology consists of the following separate yet interdependent stages (Figure 3.3).

- i). Literature review – to recognise the theories that will guide this study and to identify gaps in the literature.
- ii). Create a catalogue of fraud symptoms (critical combinations and known fraud symptoms).
- iii). Identify data requirements to detect fraud in enterprise systems, in general and SAP, in particular.
- iv). Design, develop and implement prototype software on a stand-alone computer system.
- v). Perform experiments with simulated test data and modify prototype to ensure it produces correct results.
- vi). Perform experiments with case study data to obtain proof that the prototype can detect potential fraud in a real organisation.
- vii). Seek support validating the prototype.

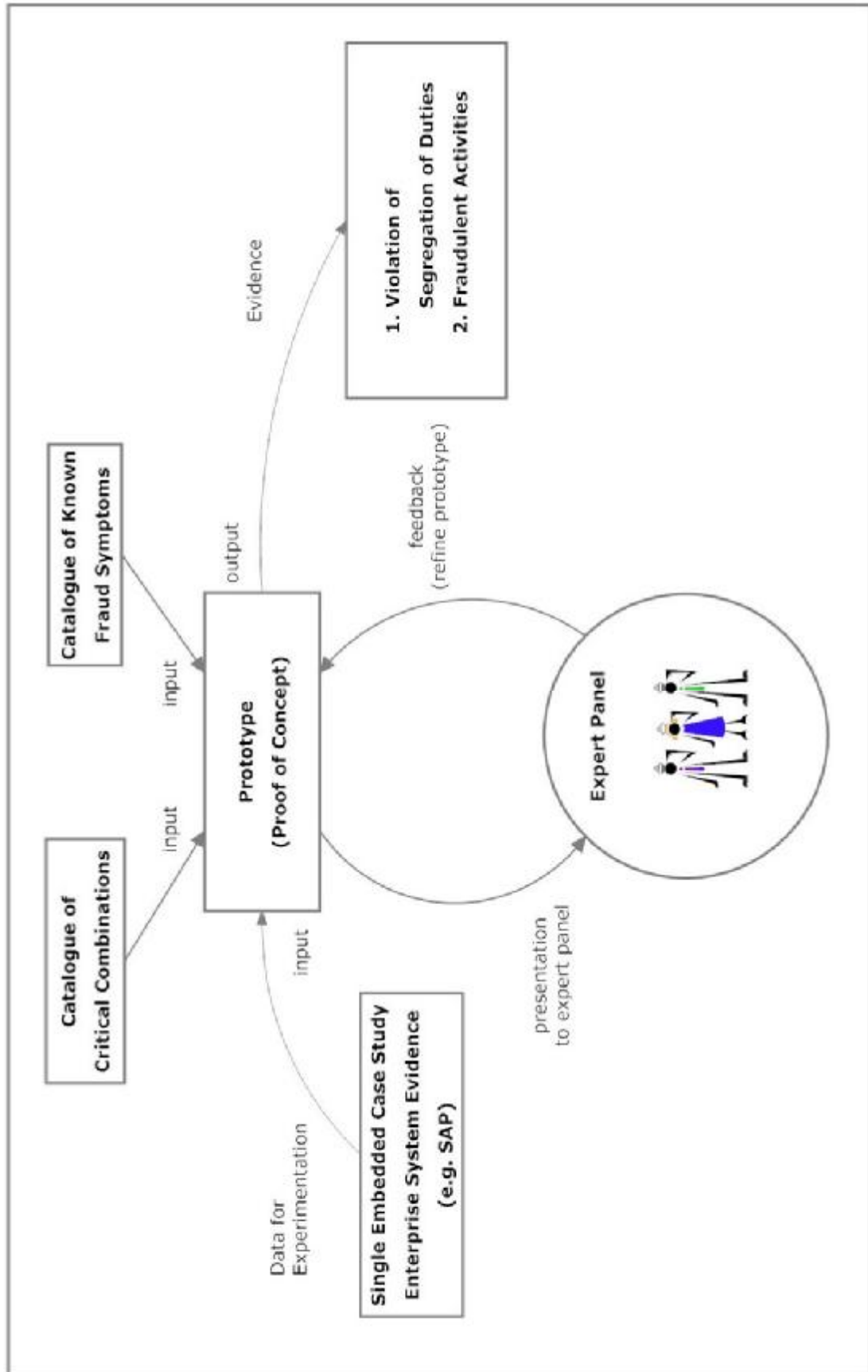


Figure 3.3: Methodology

The concept of proactive detection of potential fraud in enterprise systems is demonstrated by developing a prototype. The aim of the prototype is to confirm the feasibility of implementing proactive detection of potential fraud in practice. It is a software application that performs data analytics on transaction data obtained from a SAP enterprise system. Reports and visualisations indicating potentially fraudulent or anomalous activities are produced as output.

3.3.1. Scope of fraud categories

It was noted in the literature review that 'asset misappropriation' is the most common category of fraud in organisations. Furthermore, non-management employees are the largest group involved in perpetrating this fraud against organisations. These employees used billing schemes primarily to perpetrate fraud against organisations. Two of the most widely exploited weaknesses are the lack of internal controls and overriding of existing internal controls, resulting in a violation of segregation of duties.

When considering an automated solution for proactive detection of potential fraud, the focus has to be on questions that can be answered with the aid of computerised tools (Lanza 2007). Some questions are too subjective, for example, "Are the vendor's goods or services of good quality?" Any effort to develop an automated solution will require evidence that: i) is documented in an enterprise system's audit trails; and ii) can be investigated using data analytics tools. Transactions that occur outside an enterprise system cannot be investigated using this methodology.

The focus of this research is on 'asset misappropriation' in general, and billing schemes, namely i) shell company; and ii) non-accomplice vendor schemes within

accounts payable, in particular. Hereinafter these billing schemes are collectively referred to as vendor fraud. These schemes are among the most costly of all asset misappropriation schemes in organisations (ACFE 2010 ; Wells 2008). Furthermore, these schemes leave 'footprints' in audit trails of an enterprise system that can be detected using appropriate computerised analytics techniques. The measures used in this study to detect potential fraud will therefore focus on these schemes.

3.3.2. Measures to detect fraud

Methods developed in this study to detect fraud are based on prior work conducted by several researchers (ACFE 2010 ; Albrecht et al. 2009 ; Singleton et al. 2008 ; Wells 2008 ; Lanza 2007 ; O'Gara 2004 ; Greene 2003a ; Lanza 2003 ; Little and Best 2003). These methods are grouped into two categories: i) critical combinations; and ii) known fraud symptoms. They are used in the development of prototype software for detection of potential fraud.

Critical combinations

Many frauds occur because fraudsters exploit the lack of internal controls or they may override existing internal controls that are poorly implemented. The fraud detection prototype aims to detect user activities that violate segregation of duties. Further analysis will then be undertaken to identify any anomalous symptoms that arise due to these violations. For example, an employee that creates or modifies a vendor master record should not be able to enter an invoice. Having this capability does not indicate that a fraud has taken place, but it does create an opportunity for a fraud to be perpetrated. By detecting these critical combinations of user activities: i) an auditor can further investigate transactions that match known fraud symptoms, or

appear otherwise anomalous; and ii) an organisation can take steps to correct the situation thereby reducing the possibility of future fraud. The concept of separating critical business activities in order to reduce fraud is termed 'segregation of duties'. In its simplest form, the segregation of duties (SoDs) principle states that sensitive tasks should be divided into two or more steps with each step being performed by a different user. SoDs reduces conflicts of interest and enables detection of user activities that result in a breach of its principles (Best et al. 2009 ; Coleman 2008 ; Li et al. 2007 ; Srinidhi 1994).

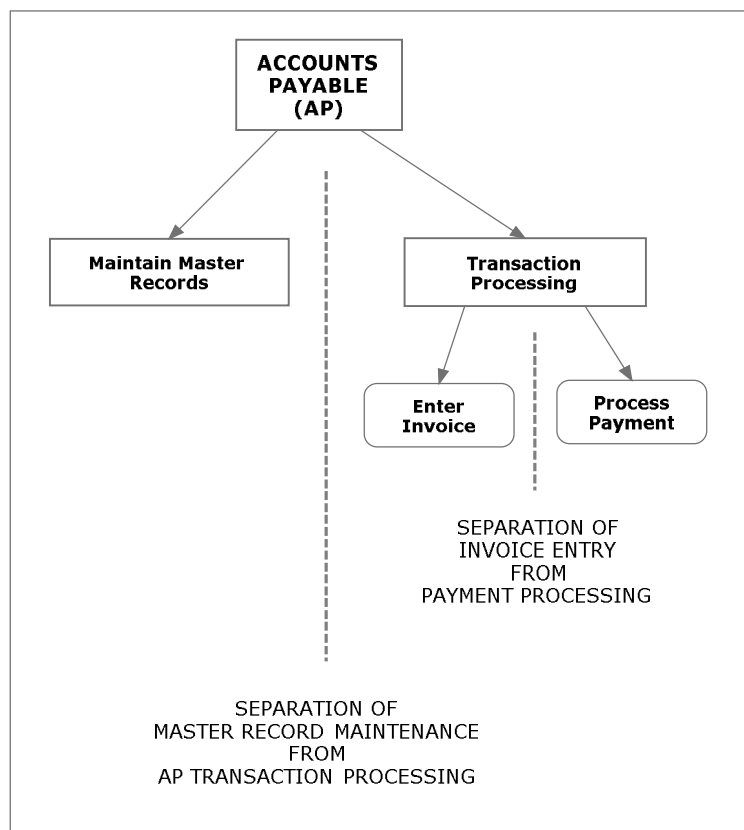


Figure 3.4: Critical AP activities model

Source: adapted from Little and Best (2003)

This study supports the following principles of segregation of duties within an accounts payable (AP) function as proposed by Little and Best (2003) (Figure 3.4):

- i). users who can create and modify master records should not be able to post transactions; and
- ii). payments should be performed by someone other than the person who enters vendor invoices.

Known fraud symptoms

Billing fraud schemes occur when a fraudster causes an organisation to issue a payment by submitting invoices for fictitious goods or services, inflated invoices, or invoices for personal purchases (these were discussed in section 2.5). The prototype will identify user activities that violate segregation of duties. The following analyses may be performed to identify anomalous user and vendor transactions:

- i). *bank account flipping* – checks for changes to banking details, a change back to original, with transactions processed in the interim period;
- ii). *duplicates testing* - checks for any duplicates, for example, invoices, payments or vendors;
- iii). *trend analysis* - compares activities over two or more periods, to identify variances over time, for example, vendors with minimal payments in prior periods but large payments in current period may be fraudulent payments;
- iv). *Benford's Law* - gives expected frequencies of digits in numerical data; spikes may be indicative of fraud and require further investigation;
- v). *stratification* - identifies the number and dollar value of vendor payments that occur within a specified interval, for example 5% below an approval limit; and
- vi). *graphing* - provides a visual means of documenting anomalous activities.

A sample of methods to detect symptoms of known shell company and non-accomplice vendor schemes is further described in Table 3.2 (Lanza 2003; Wells 2008; Best et al. 2009).

3.3.3. Data requirements to detect potential fraud

To detect potentially fraudulent activities in an enterprise system, some fundamental data is required. At a minimum to detect fraud schemes listed in Table 3.2, an MCL-based application will require access to generic data items that define the event (who, when, where, and how) as well as specific data items relating to each scheme. Accordingly, this data should minimally include:

- i). *user name* – name of the user that performed the transaction;
- ii). *date* – that the transaction was performed;
- iii). *time* – that the transaction was performed;
- iv). *computer workstation* – that the transaction was performed on;
- v). *transaction performed* - the specific transaction that the user performed (e.g. entering an invoice, posting a payment); and
- vi). *transaction details* – data relating to the transaction performed (e.g. vendor bank details, invoice amount).

Furthermore, historical data is required to detect frauds that are perpetrated over a period of several days, weeks, or months. An example of such a scenario is a change in vendor payment details followed by a change back to the original after a period of time has elapsed, with payment(s) made in the interim period (Figure 3.5).

Table 3.2: Methods to detect known fraud symptoms

Method	Purpose	Category
Extract all vendors that have a change in payment details followed by a change back to the original after a short time (flipping) with payment(s) made in the interim period	Fraudster may use legitimate, fake or duplicate invoices to redirect payments to a personal bank account by temporarily changing vendor payment details	Non-Accomplice Vendor
Perform trend analysis of vendor payments	Identify vendors with minimal purchases in prior periods but large payments in current period as these may be fraudulent payments	Non-Accomplice Vendor
Stratify vendor payments on approval limits (e.g. many \$999 payments when approval limit is \$1 000)	These payments may be an attempt to bypass review by management	Shell Company
Identify duplicate payments	Duplicate payments on vendor, invoice or amount	Shell Company
Extract vendors having different payment details	Same invoice and amount but payment to different vendors Same vendor and invoice but different amounts paid	Non-Accomplice Vendor
Identify multiple vendors sharing same payment details	Payments to same vendor using different payment details is indicative of fraud Payments to multiple vendors using same payment details is indicative of fraud	Shell Company Non-Accomplice Vendor

Source: adapted from (Lanza 2003 ; Wells 2008 ; Best et al. 2009)

... continued on next page

...Table 3.2 continued from previous page

Method	Purpose	Category
List vendors that become active after long periods of being dormant	Could indicate that employee is fraudulently using a dormant company	Shell Company Non-Accomplice Vendor
Extract all invoices with round dollar amounts	These invoices have a higher likelihood of being fraudulent	Shell Company Non-Accomplice Vendor
Extract vendors with the payments exceeds the last largest payment by a significant amount e.g. 200%	Large payments are unusual and may indicate fraud, especially when analysed in relation to other vendors	Shell Company Non-Accomplice Vendor
Extract vendors with similar names	These vendors are more prone to abuse or they may be fictitious	Shell Company
Benford's Law	Gives expected frequencies of digits in numerical data. Spikes may be indicative of fictitious invoice or payments.	Shell Company Non-Accomplice Vendor

Source: adapted from (Lanza 2003 ; Wells 2008 ; Best et al. 2009)

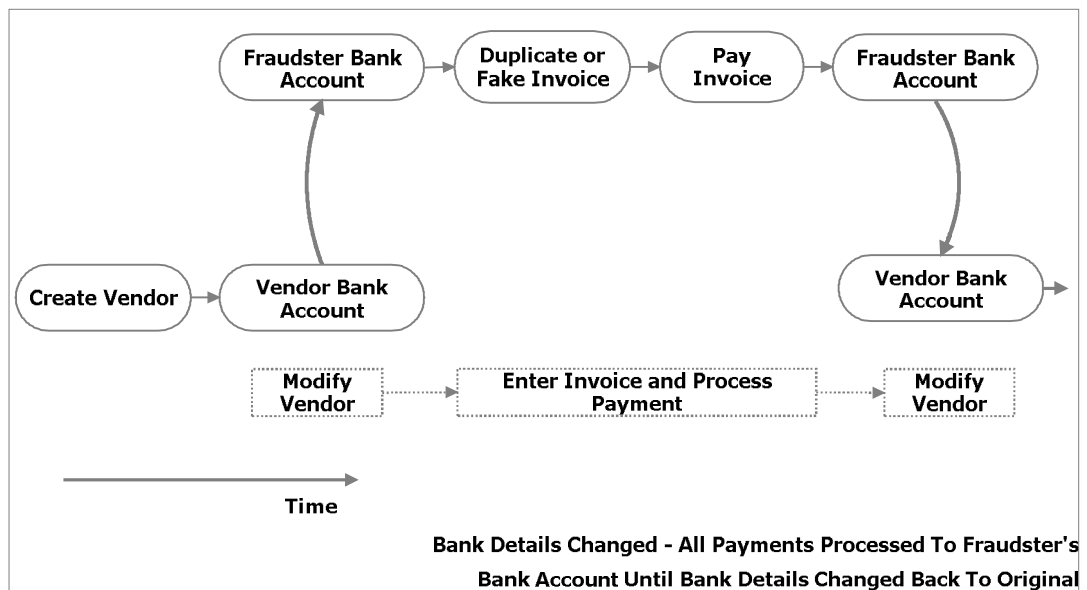


Figure 3.5: Flipping vendor bank account details

In this scenario a fraudster uses legitimate, fake or duplicate invoices to redirect payments to a personal bank account by temporarily changing a vendors payment details. In order to effectively detect this fraud scenario, data spanning the duration of the fraud is required.

3.3.4. Prototype

A prototype is a partial or simplified implementation of a complete system (Davis 1992 ; Asur and Hufnagel 1993) built for a specific purpose such as:

- i). formulating and evaluating requirements, specifications and designs;
- ii). demonstrating feasibility, system behaviour or performance;
- iii). identifying and reducing risks of system mis-development;
- iv). communicating ideas, especially among diverse groups; and

- v). answering questions about specific properties of proposed systems.

(Luqi and Steigerwald 1992)

Prototyping applies to all kinds of systems, such as software, hardware, people or a combination thereof. A prototype is a low cost representation of a system that displays selected aspects of the proposed system. Within the information systems domain, prototypes provide: i) an operational model of the application that implements certain aspects of the future system; ii) a concrete basis for discussions between developers, users and management; and iii) a basis for subsequent prototypes or for the application system. Prototypes may also be used: i) to clarify any relevant specification or development problems; ii) to serve as a basis for discussion and as an aid to decision-making; and iii) for experimental purposes and for gaining practical experience (Budde and Zullighoven 1990).

Two key advantages for constructing software prototypes that are relevant to this study are: i) to provide users with a 'tangible' idea of the problem solution being sought after; and ii) to demonstrate the technical feasibility of a specification (Asur and Hufnagel 1993 ; Budde and Zullighoven 1990).

There are four steps in developing a software prototype (Figure 3.6).

- i). Identify a user's needs - meet with users to agree on what elements the system should include and exclude. The emphasis is on *what* the system should produce and not on *how* it should be produced.

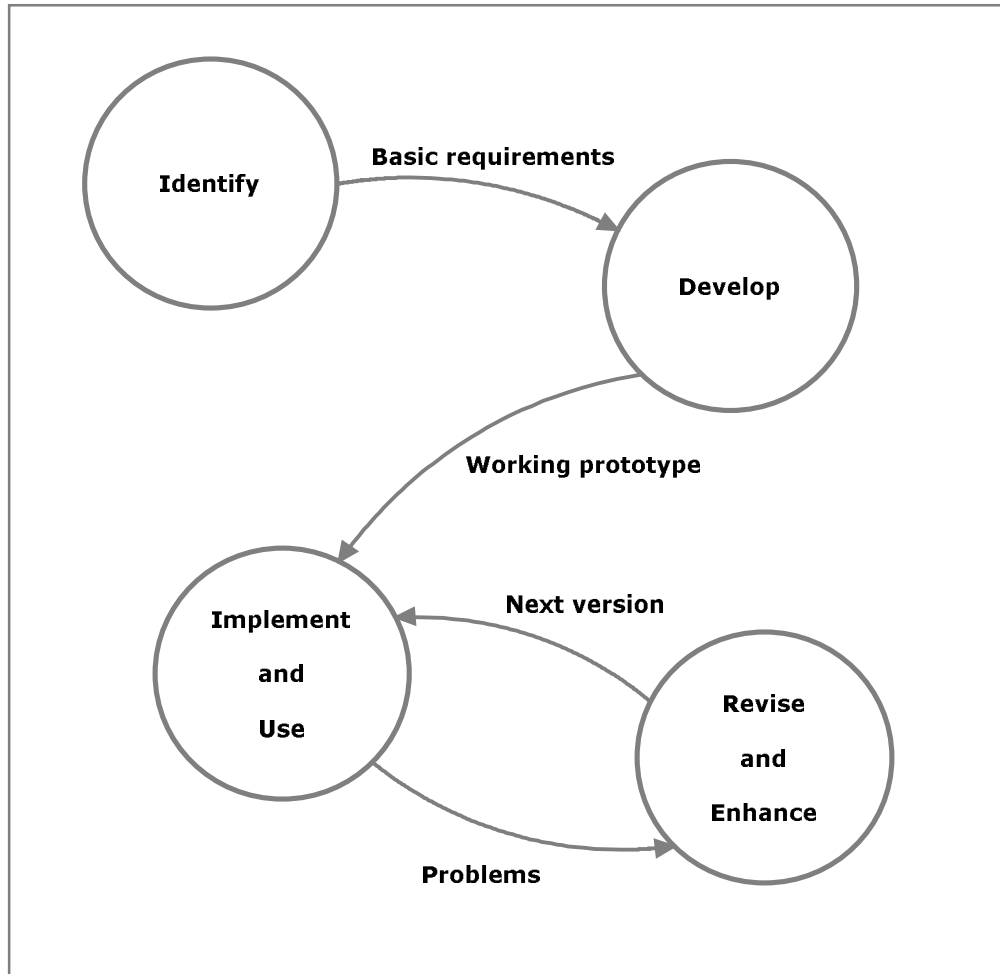


Figure 3.6: The prototype model

Source: (Naumann and Jenkins 1982 p.31)

- ii). Develop a working prototype –initial prototype is built in a short time, meeting the agreed-on user's requirements. The goal of this stage is speed; non-essential aspects of the system are not included in this stage.
- iii). Implement and use the prototype – 'hands-on' use of the system. Users are provided with tentative versions of data entry screens, menus, prompts and documents. They are also required to respond to prompts, query the system, judge response times and issue commands.

- iv). Revise and enhance the prototype - any undesirable or missing features are identified and corrected.

(Romney and Steinbart 2009 ; Asur and Hufnagel 1993 ; Naumann and Jenkins 1982)

This study proposes a two phase process for automated detection of potential fraud using a prototype. In phase one, data is extracted from a SAP enterprise system. Phase two is sub-divided into two stages. In stage one, surveillance of data is conducted to identify user activities that violate segregation of duties. In stage two, user activities that violate segregation of duties are investigated in detail to identify transactions that match known fraud symptoms, or appear otherwise anomalous (Figure 3.7). The prototype utilises a *catalogue of critical combinations* and *known fraud symptoms* (section 3.3.2) to detect potentially fraudulent transactions. Anomalous and potentially fraudulent activities are identified; and reports and visualisations are produced.

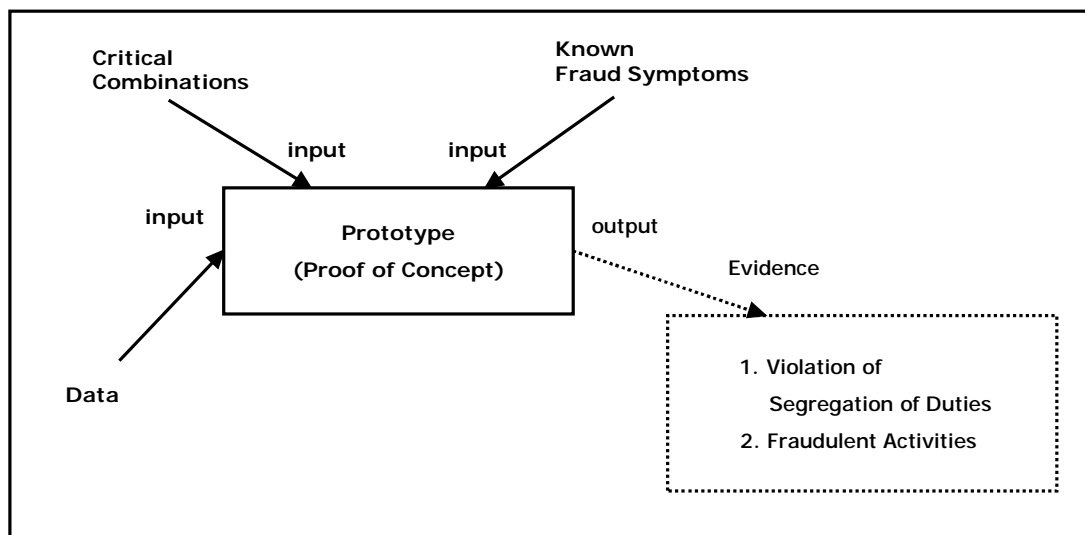


Figure 3.7: Prototype input requirements

The prototype provides evidence in support of the concepts proposed in the research. It allows the research to make the statement "*...and it is feasible to practically implement proactive detection of potential in enterprise systems.*"

Test Data

Test data covering a period of one (1) month is generated to simulate user activities in a SAP enterprise system. The following activities are embedded in the test data.

- i). User activities - simulated user activities include vendor maintenance; invoicing; and payment transactions.
- ii). Critical combinations - user activities violating SoDs principles, namely, users creating and modifying master records and posting transactions; and users entering invoices and processing payments.
- iii). Anomalous user transactions - 'flipping' of vendor bank account details while payments are processed in the interim period; duplicate invoices and payments; round dollar invoices and payments.
- iv). Anomalous vendor transactions - vendors with similar names; vendors posting transactions after being dormant for long periods; vendors sharing bank accounts; vendors having multiple bank accounts; and vendors having multiple master records.

A series of 'manual' experiments are performed on the test data to establish a set of 'control' values using Microsoft Excel. The same experiments are conducted with the prototype and 'experimental' values produced are reconciled with 'control' values. Inconsistencies in results are used to correct errors in the prototypes syntax, program logic, and knowledge base.

3.3.5. Data collection

Data is obtained from a single embedded case study. Audit trails are extracted from a SAP enterprise system and exported to a separate personal computer system for analysis. These audit trails routinely capture activities that are performed within a SAP enterprise system. Evidence supporting the primary research question is obtained by *analysis of archived enterprise system audit trails*.

Case study

"A case study is an empirical study that investigates a contemporary phenomenon in depth and within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident" (Yin 2002 p.13).

The case study is an ideal method when an in-depth investigation of a real-life phenomenon is required (Yin 2002 ; Feagin et al. 1991). It is the preferred method when the research wants to deliberately investigate contextual conditions that may be pertinent to the phenomenon being studied (Yin 2009).

There are four types of designs for case studies - single-case holistic designs, single-case embedded designs, multiple-case holistic designs, and multiple-case embedded designs. Prior to data collection a researcher needs to decide between using a single or multiple case design to address research questions (Yin 2002).

This study uses a single-case embedded design. The rationale is that transaction data from the same single case is examined to determine how it changes over time, i.e. a *longitudinal* case study. An embedded design uses multiple units of analysis. In this

study, the main unit of analysis is an organisation and intermediary units are employees and vendors. This design is justifiable because data from the same case will be investigated to determine how it changes over time and whether these changes are indicative of potentially fraudulent activities.

There are at least six sources of data when using case studies: i) documentation; ii) archival records; iii) interviews; iv) direct observation; vi) participant observation; and vii) physical artefacts (Yin 2002 ; Stake 1995). The primary source of data for this study is '*archival records*'. These archival records are in the form of audit trails of transactions processed in an enterprise system (Figure 3.8).

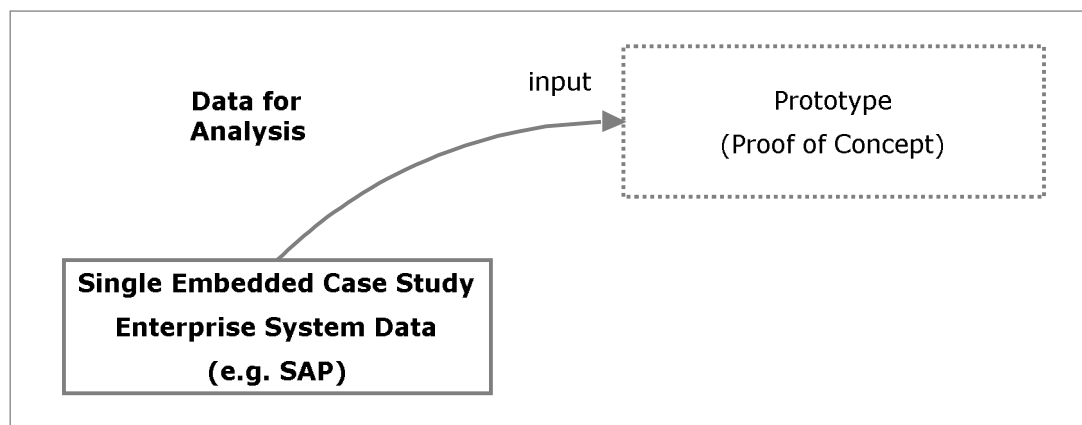


Figure 3.8: Source of data

3.3.6. Proof of conceptual model

Evidence supporting the research is obtained by analysing data obtained from an enterprise systems audit trails. Analysis is performed on the data in two stages. In stage one, the data is analysed to determine if any critical combinations of user activities are present. These activities violate segregation of duties and require further investigation. In stage two, user activities that violate segregation of duties

are investigated in detail to determine whether they match known fraud symptoms, or appear otherwise anomalous. Each of these stages is discussed below.

Stage one analysis

The purpose of this stage is to identify any critical combinations of user activities. These activities violate segregation of duties and need further investigation. The activities include:

- i). users who can create and modify master records should not be able to post transactions and;
- ii). in accounts payable, payments should be performed by someone other than the person who enters vendor invoices.

Stage two analysis

The purpose of this stage is to identify potentially fraudulent transactions that match known fraud symptoms, or are otherwise anomalous. In this stage analyses for known fraud symptoms described in section 3.3.2 and Table 3.2 are performed.

3.3.7. Expert panel validation of model

A panel of experts are requested to observe and provide feedback on performance of the prototype. Their feedback is obtained on issues such as *operation, reporting and visualisations, accuracy and efficiency, and impact on auditor productivity*. This feedback provides evidence demonstrating validation and acceptance of the prototype.

An expert panel consists of a group of selected individuals that have extensive skill or knowledge in a particular field (Collins 2000 ; Fuller 2002). An expert panel

provides the researcher with an opportunity to discuss issues and to get expert feedback on issues relevant to the research topic. The research interest directs the discussion while data comes from interactions between members (Morgan 1997). A researcher takes on the role of moderator in these interactions (Parker and Tritter 2006). These sessions provide an efficient means of observing a large amount of interaction on a topic in a limited amount of time. Interaction produces data and insights that would otherwise be less attainable in the absence of such an interaction. (Farnsworth and Boon 2010 ; Morgan 1997).

Expert panels are intended to elicit in-depth information, viewpoints, and opinions from experts (Moy 2008). Panel members are independent experts, recognised in the field of study being evaluated. They examine all the information on the topic and provide their expert feedback and answers to evaluative questions. The panel does not fully explain its judgement, but credibility of the evaluation is guaranteed by the fact that conclusions result from consensus between members that are specialists in their field (EC 2011).

Participants in an expert panel are selected based on the purpose of the study (Robinson 1999). Random selection is not critical because the intent is to understand how experts in the panel think and talk about the topic being studied (Krueger and Casey 2000). It is therefore important to recruit participants that have experience and interest in the topic being studied. The terms of reference given to the panel may include presentations, questions, and an estimation of real or probable effects (EC 2011). Homogeneity of participants (for example, similar backgrounds and

experiences) is recommended (Plummer-D'Amato 2008) as randomly selected participants may not be interested in research.

Expert panels require a central topic. In order to direct interactions it is essential for a moderator to prepare a series of questions to be answered by panel members. The process of reaching conclusions by an expert panel involves the following stages. Firstly, presentation of the research and any results or questions to be answered. Secondly, experts, either individually or in small groups, conduct their own examination of the research. Finally, an expert panel provides their interpretations and findings in a balanced and impartial way. This may be recorded in a survey.

In the context of this study, an expert panel consisting of 20 experts in the domain of auditing is used. All members of CPA Australia (Queensland Division - IT Discussion Group) and ISACA³ (Queensland Chapter) are invited to participate in the expert panel. A short presentation on the research topic and demonstration of the prototype is given (20 minutes) and members have a hands-on session using the prototype. Their feedback is sought using a survey (Appendix 3) on the following key issues, namely *operation, reporting and visualisations, accuracy and efficiency, and impact on auditor productivity*. Feedback from the panel is used to validate the prototype (Figure 3.9).

Expert panels are useful to explore participant's thoughts, ideas and experiences in relation to a topic of study. To maximise success the researcher must carefully consider the composition of the panel and be suitably prepared with a series of

³ Information Systems Audit and Control Association

questions to ensure that discussion progresses naturally towards answers being sought. The expert panel protocol for this study is shown in Appendix 2.

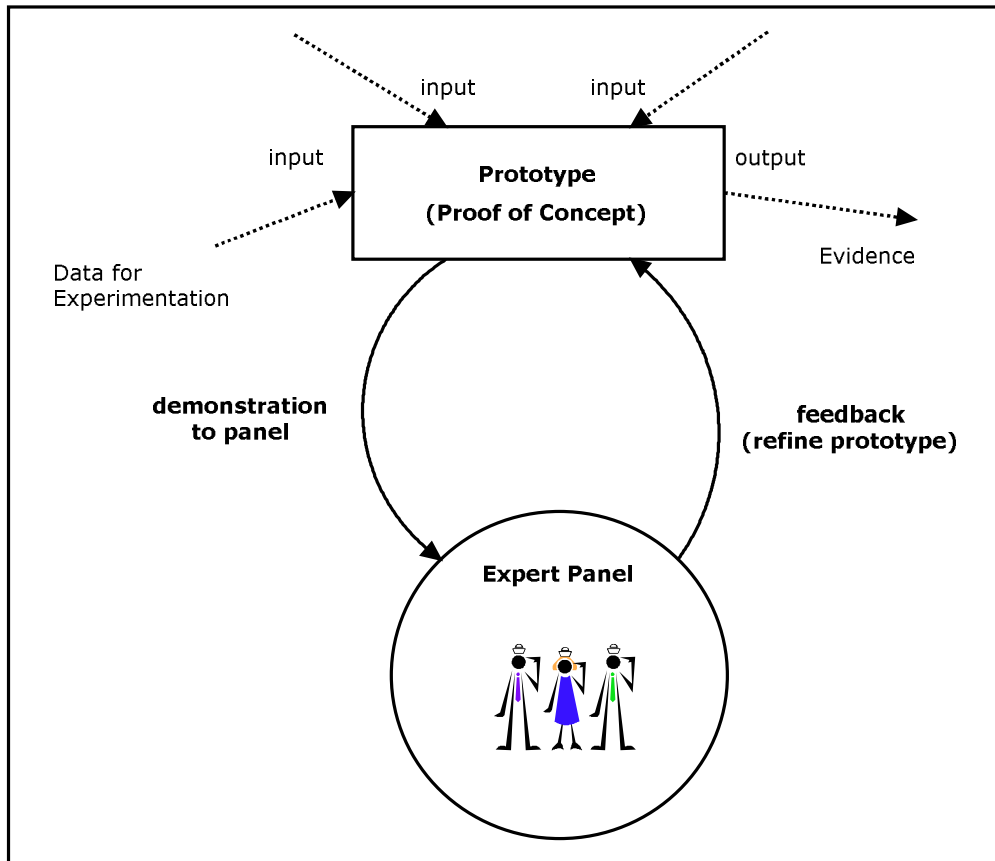


Figure 3.9: Expert panel interaction and feedback

3.4. Conclusion

This research aims to answer the primary research question whether *"a generalised model for proactive fraud detection in enterprise systems can be developed"*. The approach to answering this question is to develop a prototype. The prototype provides support for the statement *"...and it is feasible to practically implement proactive fraud detection in enterprise systems."* The prototype analyses data from audit trails of a SAP enterprise system. A catalogue of *critical combinations* of user activities and *known fraud symptoms* are used in conjunction with various analysis

techniques to detect and report potentially fraudulent symptoms. Evidence supporting the research is obtained as output from the prototype. Feedback is sought from an expert panel to establish validity of the prototype. Results from data analysis provide further support for the prototype.

CHAPTER 4

Prototype Design

4.0. Introduction

This research proposes that a generalised model for proactive detection of potential fraud in enterprise systems can be developed. In order to achieve this outcome a prototype is developed. The aim of the prototype is to test the feasibility of implementing proactive fraud detection in practice. The prototype analyses data extracted from a SAP enterprise system to identify user activities that violate segregation of duties. Users and vendors may be further investigated at the discretion of an auditor to identify activities that match known fraud symptoms, or appear otherwise anomalous. A catalogue of fraud symptoms informs this process.

The previous Chapter discusses the research design and methodology used in this study. In this Chapter design specifications for the prototype is produced. This Chapter focuses on:

- i). identifying data required to detect fraud in enterprise systems, in general, and SAP in particular;
- ii). creating a catalogue of fraud symptoms, i.e. *critical combinations* of user activities, and *known fraud symptoms*; and
- iii). designing strategies to detect fraud.

This Chapter also addresses the following three research propositions.

- § **RP1a:** Enterprise system audit trails document adequate data to allow retrospective monitoring of user activities.
- § **RP1b:** Violations in segregation of duties can be identified by analysing audit trails for critical combinations of user activities.
- § **RP1c:** Potentially fraudulent transactions can be identified by investigating user activities that violate segregation of duties, match known fraud symptoms, or appear otherwise anomalous.

The following sections provide a detailed discussion of the prototype design.

4.1. Prototype design

The prototype consists of four interrelated modules (Figure 4.1).

- i). *Input* - data extraction and preparation.
- ii). *Process* - fraud detection engine – pre-processing data for fraud symptoms and production of reports and visualisations.
- iii). *Storage* – of input and historical data and data tables produced during pre-processing.
- iv). *Output* - display reports and visualisations in a web-based interface.

Data is extracted from an organisation's SAP enterprise system and prepared by the *input* module for incorporation into storage. This module specifies data requirements for fraud detection. The *process* module is the fraud detection engine. It analyses transaction data and reports on fraudulent activities that have occurred. Three

essential elements make up this module; a catalogue of critical combinations, a catalogue of known fraud symptoms, and fraud detection strategies. The *storage* module stores data required for fraud detection, including input data, historical data and storage of data tables. The *output* module provides summarised and detailed reports and visualisations.

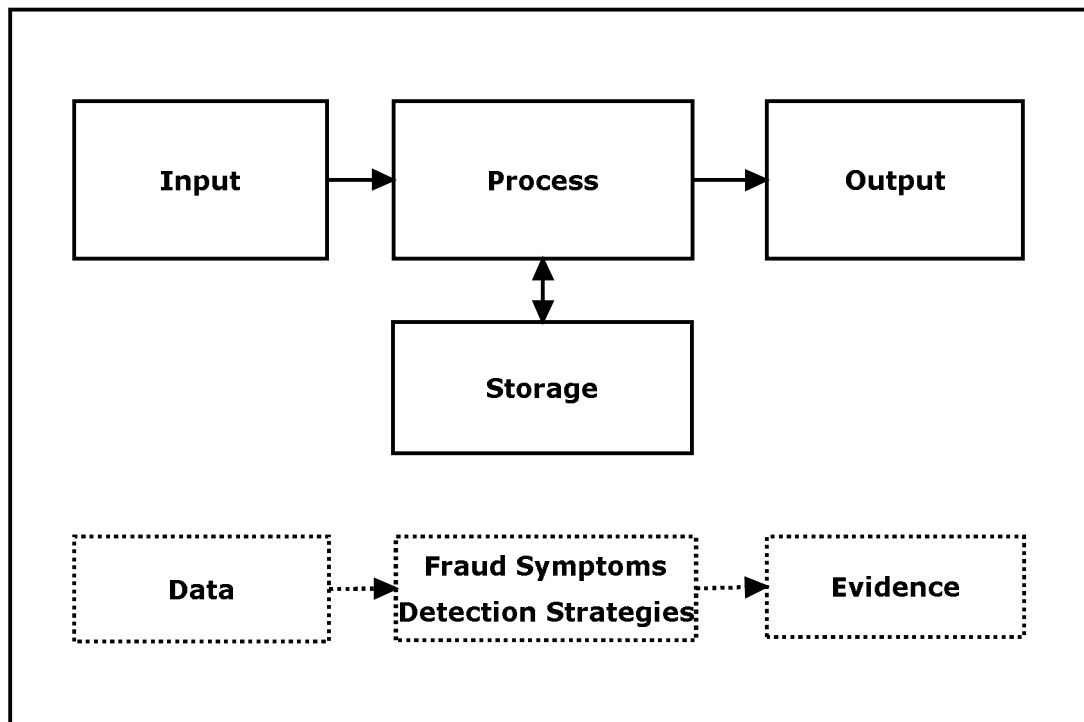


Figure 4.1: Prototype conceptual design

The requirements and design specifications for each of the modules is discussed in the following sections.

4.2. Data requirements for fraud detection

Fundamental audit trail data required to detect fraud symptoms in an enterprise systems is discussed in Chapter 3. In this section, fraud detection methods developed

in Chapter 3 are examined in detail to identify data sources, and data extraction requirements for a SAP Enterprise System (Figure 4.2 and Table 4.1).

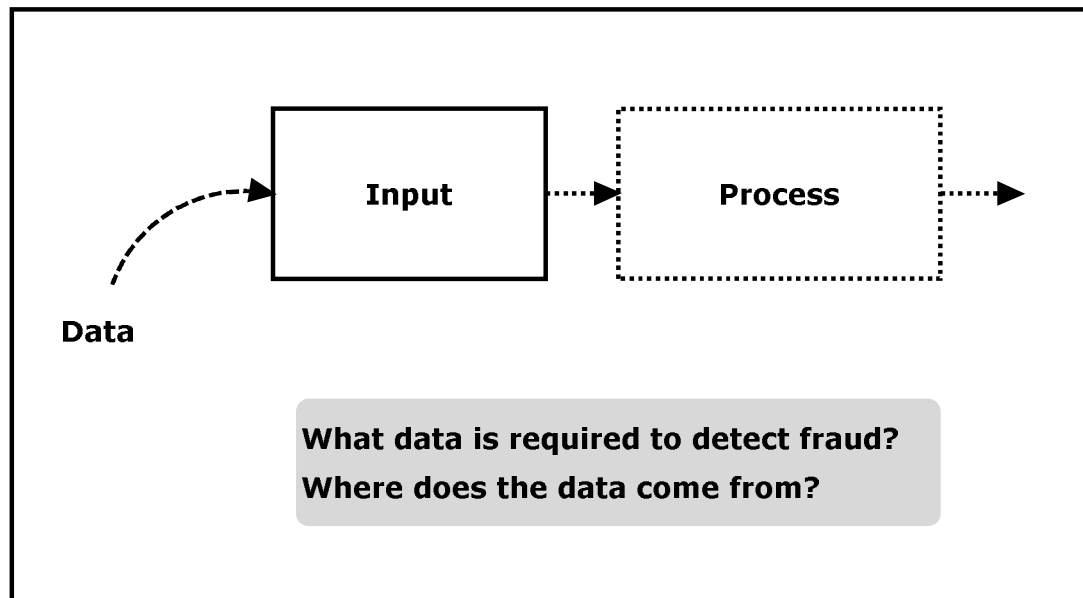


Figure 4.2: Input specifications

The principal sources of data for detection of fraud symptoms in enterprise systems, in general, are:

i). vendor master file

- § vendor name
- § street address
- § P.O. box address
- § telephone
- § tax number
- § bank BSB
- § bank account
- § created/ modified by
- § creation date
- § creation time

ii). invoice payment file

- § vendor name
- § invoice number
- § invoice date
- § posting date
- § payment date
- § amount
- § user name

iii). employee master file (optional)

- § employee first name
- § employee last name
- § telephone
- § address
- § bank BSB
- § bank account

The employee master file is optional. The rationale is that enterprise systems such as SAP are modular in structure. As a result of this modular structure organisations may choose not to implement the human resources (HR) module. In this situation an employee master file is not available in an enterprise system. Although an employee master file may or may not be available, user names of employees are documented in the vendor master and invoice payment files. This may be used to identify users that perform suspicious activities. A SAP enterprise system is examined next to determine the level of support it provides for fraud detection.

4.3. SAP support for fraud detection

A SAP enterprise system provides audit trails that, in the context of fraud detection, may be used to monitor user activities. SAP audit trails consist of security audit logs,

records of changes to master records and, accounting audit trails (Best et al. 2009). The following discussion examines how these audit trails may support fraud detection.

4.3.1. SAP audit trails

SAP audit trails provide detailed descriptions of functions performed within an enterprise system. Each function in SAP has a transaction code associated with it. A transaction code (or t-code) consists of letters, numbers, or both (for example, FB60 – Enter Vendor Invoice). A transaction code is a shortcut that takes a user directly to a SAP application rather than having to navigate through the menu system (Padhi 2010). Each transaction code executed by a user is recorded in an audit trail (Best 2000). Audit trail data required for this research is stored in several tables within a SAP enterprise system (Figure 4.3).

Changes to master records are stored in two tables (Figures 4.4 and 4.5) namely **CDHDR** Change Document Headers and **CDPOS** Change Document Items (Padhi 2010 ; Best et al. 2009 ; Hirao 2009 ; Best 2005). Changes to master records include creation and deletion of master records and changes to fields. Each change document header record in table CDHDR specifies: Client, Object Class of the master record, e.g. category of vendor, customer, general ledger account, cost centre, etc., Object Value, i.e. vendor number, cost centre code, Change Document Number, User Name who made the change, Date, Time, and Transaction Code (for example, FK01 - Create Vendor Master Record, FK02 - Change Vendor Master Record).

For every change document number, there is a corresponding change document item in a CDPOS table (Padhi 2010 ; Best et al. 2009 ; Hirao 2009 ; Best 2005). Change document items have the following fields: Client, Object Class (of the master record, for example, category of vendor, customer, general ledger account, cost centre, etc.), Object Value, (i.e. vendor number, cost centre code), Change Document Number, Table Name (for example, LFBK – Vendor Master Bank Details), Table Record Key, Field Name, Change Type - U(pdate), I(nsert), E(delete single field), D(elete) Record. Tables CDHDR and CDPOS are linked by the Object Id field. Thus it is possible to identify an individual user making these changes.

Table 4.1: Source of data to detect known fraud symptoms

Method	Evidence
Extract all vendors that have a change in payment details followed by a change back to the original after a short time (flipping) with payment(s) made in the interim period	Vendor Master
Perform trend analysis of vendor invoices and payments. Perform Benford's Law analysis of vendor invoices and payments	Vendor Master Invoice / Payment
Stratify vendor payments on approval limits (e.g. many \$999 payments when approval limit is \$1 000)	Invoice / Payment
Identify duplicate payments	Invoice / Payment
Extract same vendors having different payment details	Invoice / Payment
Identify multiple vendors sharing same payment details	Vendor Master
List vendors that become active after long periods of being dormant	Invoice / Payment
Extract all invoices with round dollar amounts	Invoice / Payment
Extract vendors where payments exceed last largest payment by a significant amount e.g. 200%	Invoice / Payment
Extract vendors with similar names	Vendor Master

Source: adapted from (Lanza 2003 ; Wells 2008 ; Best et al. 2009)

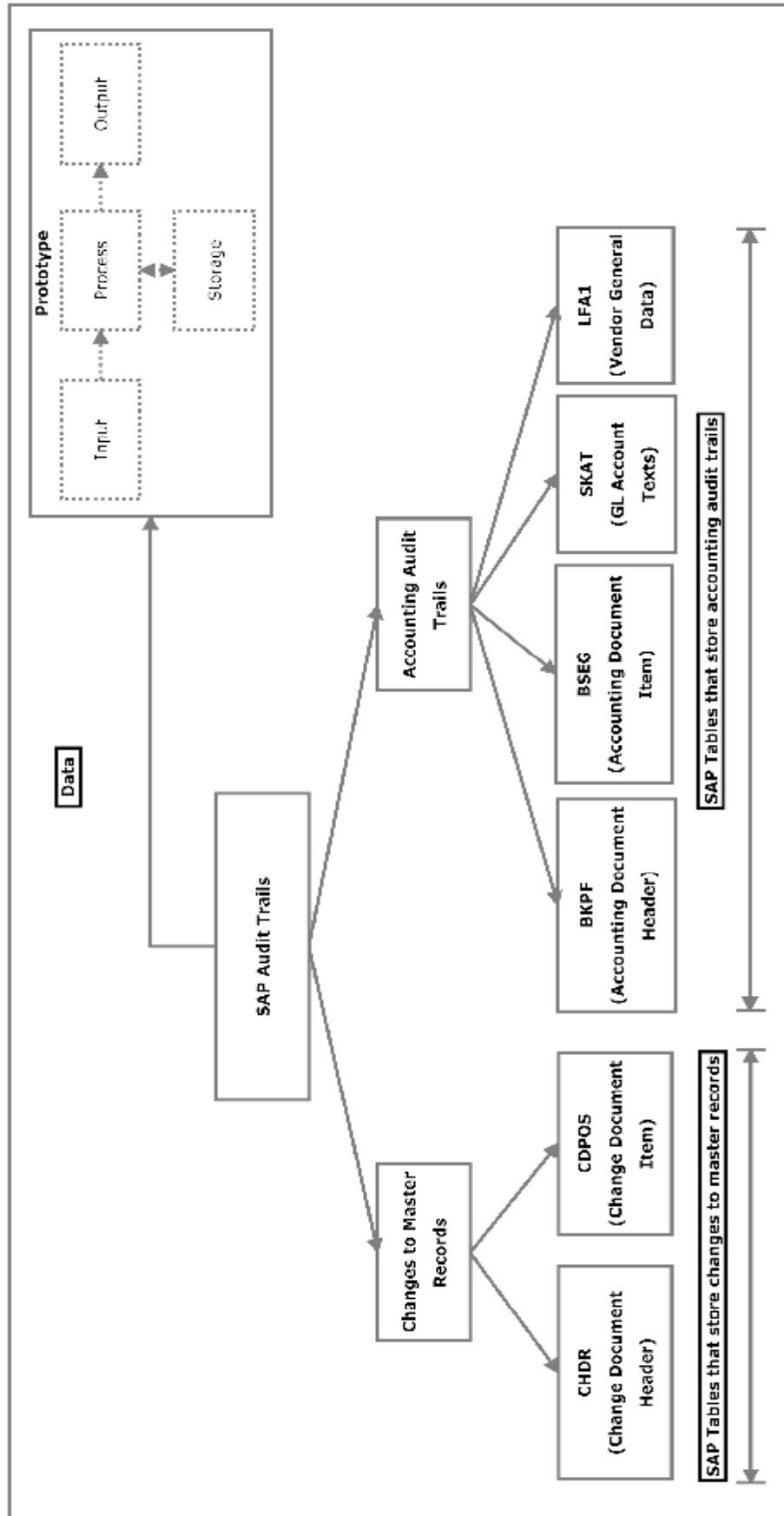


Figure 4.3: SAP audit trails

CDHDR – Change Document Headers						
Client	Object Class	Object Id	Change No.	User	Date	T-Code
110	KRED	0000111234	0000126151	GRAYW	29.03.2011	FK01
110	KRED	0000111234	0000126253	JAMESP	10.05.2011	FK02
110	KRED	0000111234	0000126354	JAMESP	13.05.2011	FK02
110	KRED	0000111235	0000126724	JOHNR	20.05.2011	FK01

Figure 4.4: CHDR – change document headers table

Source: SAP Enterprise System

CDPOS – Change Document Items						
Client	Object Class	Object Id	Change No.	Table	Table Key	Field
110	KRED	0000111234	0000126151	LFBK	1100000111234AU 232-121 1234567	KEY I
110	KRED	0000111234	0000126253	LFBK	1100000111234AU 434-545 7654321	KEY I
110	KRED	0000111234	0000126354	LFBK	1100000111234AU 232-121 1234567	KEY I
110	KRED	0000111235	0000126724	LFBK	1100000111235AU 621-543 4567891	KEY I

Table Key = Client, Vendor No., Bank Country, Bank Key, Bank Account No.

Figure 4.5: CDPOS – change document items table

Source: SAP Enterprise System

BKPF – Accounting Document Headers							
Client	Co. Code	Doc. Number	Fiscal Year	Doc. Date	Post. Date	User Name	T-Code
110	1234	100000010	2010	10.05.2011	10.05.2011	JAMESP	FB60
110	1234	100000502	2010	18.05.2011	18.05.2011	SMITHA	FB60

Figure 4.6: BKPF – accounting document headers table

Source: SAP Enterprise System

BSEG – Accounting Document Line Items							
Client	Co. Code	Doc. Number	Fiscal Year	Dr/Cr	G/L Account	Amount	Vendor No.
110	1234	100000010	2010	Dr	751000	900.00	
110	1234	100000010	2010	Dr	709300	90.00	
110	1234	100000010	2010	Cr	709395	990.00	111234
110	1234	100000502	2010	Dr	752005	621.95	
110	1234	100000502	2010	Dr	709300	62.19	
110	1234	100000502	2010	Cr	709395	684.14	111235

Figure 4.7: BSEG – accounting document line items table

Source: SAP Enterprise System

LFA1 – Vendor General Data				
Client	Vendor No.	Vendor Name	Date Created	Created By
110	111234	Toms Hardware	29.03.2011	GRAYW
110	111235	Computer Mart	20.05.2011	JOHNR

Figure 4.8: LFA1 – vendor general data table

Source: SAP Enterprise System

SKAT – General Ledger Account Texts				
Client	Chart of Accounts	G/L Account	Short Text	
110	CAAU	709395	Accounts Payable	
110	CAAU	709300	GST Paid	
110	CAAU	752005	Printer Consumables	
110	CAAU	751000	General IT Services	

Figure 4.9: SKAT – General Ledger Account Texts Table

Source: SAP Enterprise System

Accounting audit trails are stored in tables **BKPF** – Accounting Document Header, **BSEG** – Accounting Document Line Item, **SKAT** – General Ledger Account Texts, and **LFA1**- Vendor General Data (Figures 4.6 to 4.9). Tables BKPF and BSEG store posting histories for both general ledger accounts and subsidiary ledger records. This facilitates integration of data and automatic reconciliation of subsidiary ledgers with control accounts. General ledger account texts (names) are stored in table SKAT. Vendor general data including vendor name, date created and creating user are stored in table LFA1 (Best et al. 2009). Thus it is possible to identify an individual user performing these activities.

As can be noted from the preceding discussion, data describing user activities is well-documented in audit trails of a SAP enterprise system. Detecting user activities and analysing them for potential fraud, however, is a difficult task if done manually. The prototype developed in this research automates fraud detection, thereby making it feasible to monitor an organisation's business processes at an unprecedented level of detail in a near real-time basis with minimal auditor involvement.

4.4. Catalogue of fraud symptoms

The prototype's process module (Figure 4.10) uses a catalogue of fraud symptoms to potentially detect fraudulent activities. The catalogue consists of critical combinations of user activities, and known fraud symptoms. Each of these is discussed in the following sections.

4.4.1. Critical combinations

This study supports the principles of segregation of duties (SoDs) discussed in section 3.3.1. SAP audit trails provide detailed descriptions of functions performed by each user. Every SAP function has a unique transaction code (t-code) associated with it. Critical combinations may be identified by examining t-codes of functions performed by users. A list of t-codes pertinent to each of the SoDs principles is proposed in Table 4.2 (derived from SAP table TSTCT).

The prototype interrogates user activities in search of SAP t-codes matching critical combinations as shown in Tables 4.3 and 4.4. Users that perform these combinations are identified as having violated segregation of duties principles.

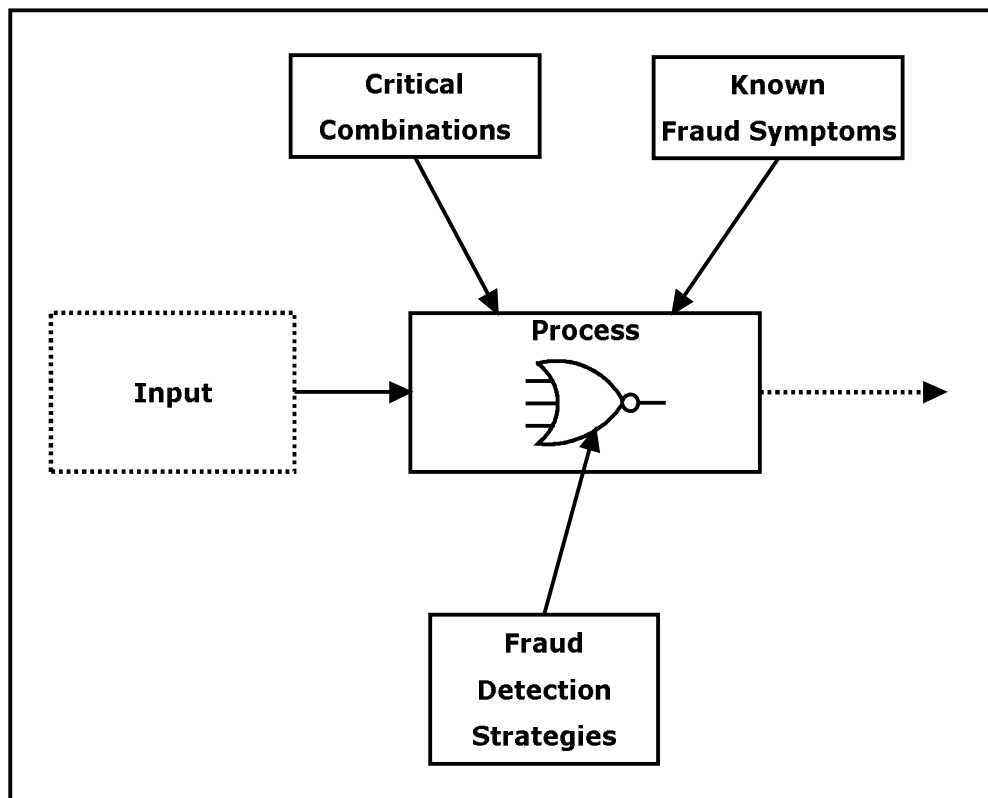


Figure 4.4: Process module

Table 4.2: SAP transaction codes

T-Code	SAP Description
Vendor Maintenance	
FK01	Create Vendor (Accounting)
FK02	Change Vendor (Accounting)
XK01	Create Vendor (Centrally)
XK02	Change Vendor (Centrally)
Enter Invoice	
FB60	Enter Vendor Invoice
F-43	Enter Vendor Invoice: Header Data
FB01	Post Document (allows posting of any financial transaction)
FB10	Invoice/Credit Memo Fast Entry
Post Payment	
F-53	Post Outgoing Payment
F-58	Post Payment with Printout
FBZ2	Post Outgoing Payments
FBZ4	Payment with Printout
F110	Automatic Payments

Source: adapted from SAP table TSTCT

SoDs Principle 1: Users who can create and modify vendor master records should not be able to post accounting transactions.

Table 4.3 lists the combination of activities a user has to perform in order to violate SoDs principle 1. If such a violation is detected then further investigation is necessary to determine whether a user has perpetrated any fraudulent transactions.

Table 4.3: Violation of SoDs principle 1

		ENTER INVOICE				POST PAYMENT				
		FB60	F-43	FB01	FB10	F-53	F-58	F110	FBZ2	FBZ4
VENDOR MAINT.	FK01	√	√	√	√	√	√	√	√	√
	FK02	√	√	√	√	√	√	√	√	√
	XK01	√	√	√	√	√	√	√	√	√
	XK02	√	√	√	√	√	√	√	√	√

Source: adapted from SAP table TSTCT

SoDs Principle 2: Payments should be performed by someone other than the person who enters vendor invoices.

Table 4.4 lists the combination of activities a user has to perform in order to violate SoDs principle 2. If such a violation is detected then further investigation is necessary to determine whether a user has perpetrated any fraudulent transactions.

Table 4.4: Violation of SoDs principle 2

		POST PAYMENT				
		F-53	F-58	FBZ2	FBZ4	F110
ENTER INVOICE	FB60	√	√	√	√	√
	F-43	√	√	√	√	√
	FB01	√	√	√	√	√
	FB10	√	√	√	√	√

Source: adapted from SAP table TSTCT

4.4.2. Known fraud symptoms

User activities that violate SoDs principles require further investigation to determine whether they match known fraud symptoms, or appear otherwise anomalous. A

catalogue of known fraud symptoms informs this process (Table 4.5). Design specifications for fraud detection strategies shown in Table 4.5 are discussed in the following section.

4.5. Design specification for fraud detection strategies

Algorithms for fraud detection are coded within the prototype's process module. They represent the core logic of the detection engine. Extracted SAP transaction data are pre-processed by the detection engine and a series of reports and visualisations are produced. The catalogue of fraud symptoms is a key component in this process (Table 4.5). In this section a comprehensive design specification, specific to the SAP Enterprise System, is produced.

Audit trails required for fraud detection were discussed in section 4.3.1. These audit trails are the basis for providing data required by the prototype. Fraud detection occurs in two phases: i) identification of critical combinations of user activities; and ii) investigation of activities that match known fraud symptoms or appear otherwise anomalous.

Critical combinations

Critical combinations of user activities were discussed in section 4.4.1. In this section algorithms to detect violation of SoDs principles are designed.

The following algorithm may be used to detect violation of SoDs principle 1 (Figure 4.4).

Table 4.5: Known AP fraud symptoms

Symptom	General Detection Strategy
Change in vendor payment details followed by a change back to the original after a short time (flipping) and payments are made in the interim period	§ Detect changes to vendor master data that result in a vendor having different bank details over a period of time. Payment of invoices is made in the interim period. The previous bank details are subsequently reinstated after being updated with new details.
Duplicate transactions	§ Check for flipping of bank details. § Check if the same payment details are used by more than one vendor.
Invoices with round dollar amounts	§ Extract all invoices with round dollar amounts (e.g. \$1000.00).
Invoices with amounts consistently below approval limit	§ Extract all vendors with multiple invoices below approval limit (e.g. several \$999 payments to vendor when limit is \$1000).
Vendors with payments exceeds the last largest payment by a significant amount	§ Extract all vendors where payment is larger than the last largest payment by a percentage e.g. 200%.
Excessive use of one-time vendors	§ Extract all payments that use the one-time account.
Vendors with similar names	§ Extract all vendors whose names are similar to other companies.
Vendors that become active after long periods of being dormant	§ Extract all vendors that become active after long periods of inactivity.
Same vendor having different payment details	§ Extract all vendors with multiple master records, each having different payment details. § Check for multiple payments using different bank account details.
Multiple vendors sharing the same payment details	§ Extract all vendors that share the same payment details.
Benford's Law (Invoices and Payments)	§ Extract all vendor invoices (or payments) and plot values against Benford's expected frequencies.

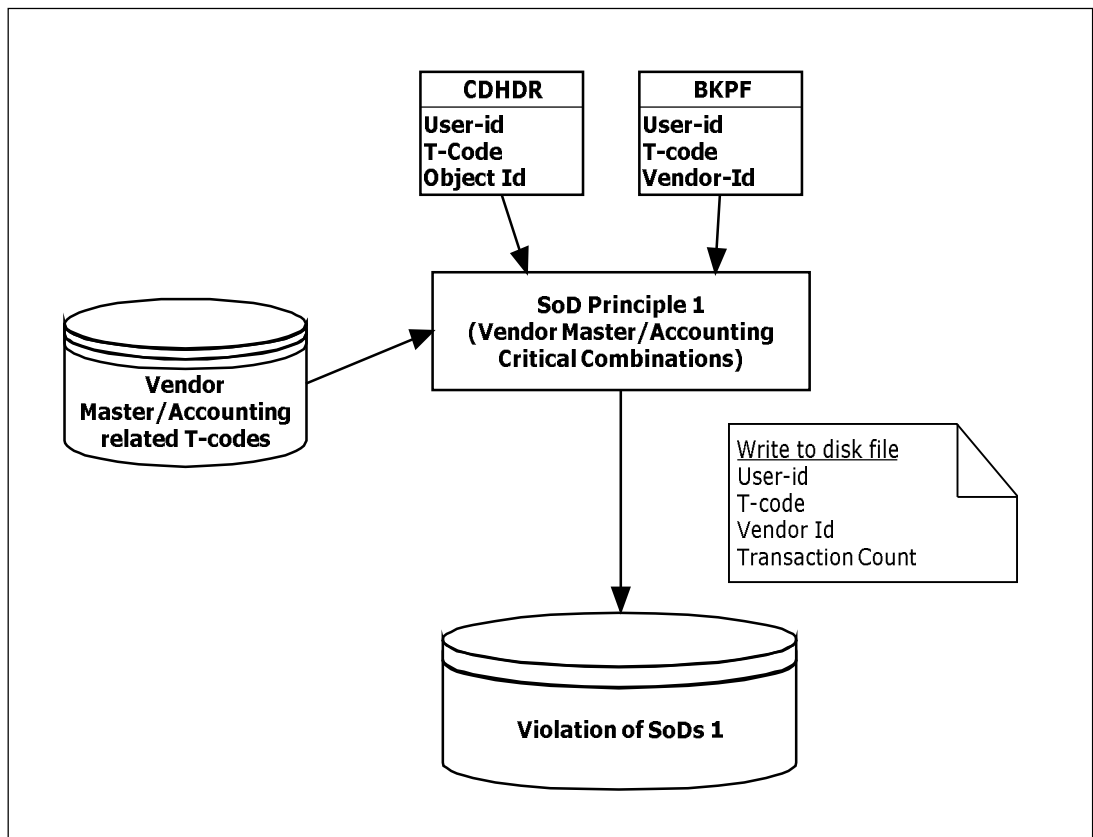


Figure 4.5: Detection strategy- users violating SoDs principle 1

```

CREATE TABLE Risky_MaintInvPmt AS
SELECT user-id, t-code, vendor-id
FROM BKPF JOIN CDHDR
ON vendor-id = object-id
AND t-code IN invoice-tcode-list
AND t-code IN payment-tcode-list
AND t-code IN vend-maint-tcode-list

```

The following algorithm may be used to detect violation of SoDs principle 2 (Figure 4.5).

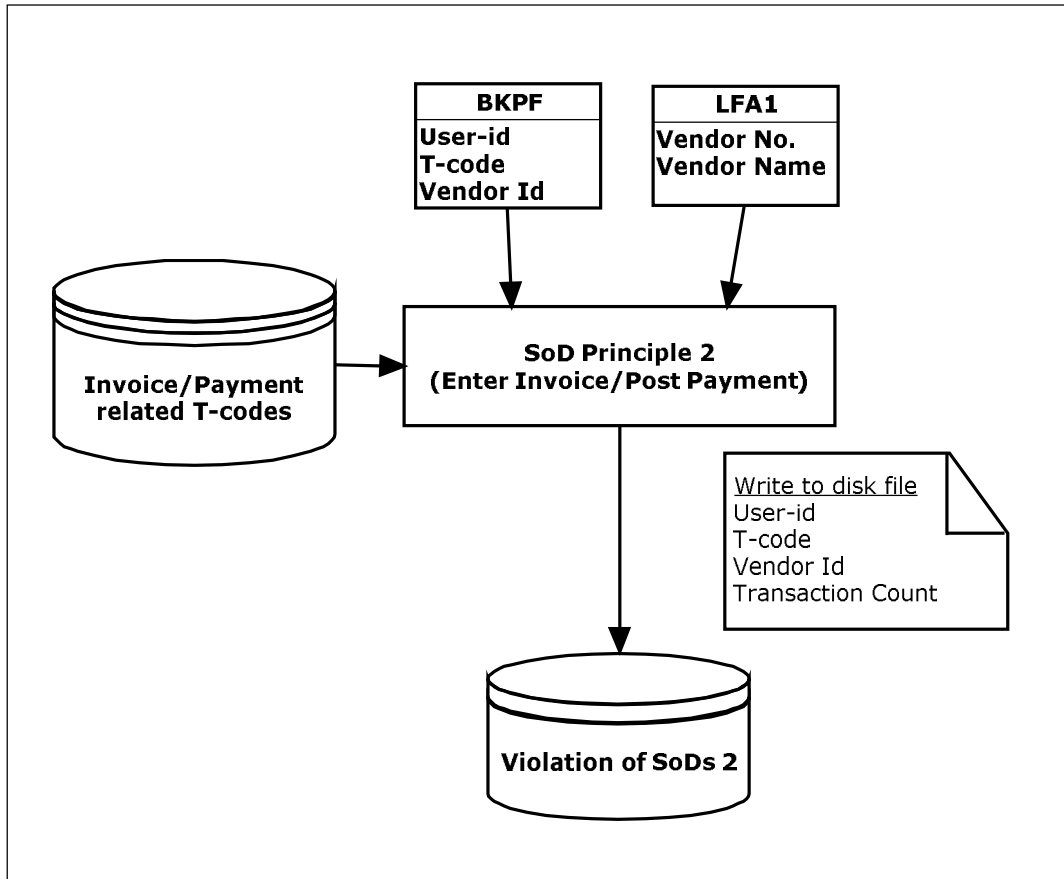


Figure 4.6: Detection strategy- users violating SoDs principle 2

```
CREATE TABLE Risky_InvPmt AS  
SELECT user-id, t-code, vendor-id  
FROM BKPF JOIN LFA1  
ON vendor-id = vendor-id  
AND t-code IN invoice-tcode-list  
AND t-code IN payment-tcode-list
```

Known fraud symptoms

General detection strategies for known fraud symptoms were discussed in section 4.4.2. In this section algorithms for detection of these symptoms are designed.

Fraud symptom 1: Flipping vendor bank account details (Figure 4.7)

```
CREATE TABLE VEND_BKPF_BSEG AS
```

```
  SELECT *
  FROM BKPF JOIN BSEG
  ON bkp.docno = bseg.docno
  AND tcode IN invoice-tcode-list
  OR tcode IN payment-tcode-list
```

```
CREATE TABLE VEND_FK_LFBK_ALL AS
```

```
  SELECT *
  FROM CDHDR JOIN CDPOS
  ON cdhdr.changenr = cdpos.changenr
```

```
CREATE TABLE VEND_HISTORY AS
```

```
  SELECT docdat as udate ,0 as utime, ,docno usnam,
         tcode, amount, vendno, " " as bkdetail
  FROM VEND_BKPF_BSEG
  UNION
  SELECT udate, utime, changenr, username,
         tcode, 0, objectid, bkdetail
  FROM VEND_FK_LFBK_ALL
```

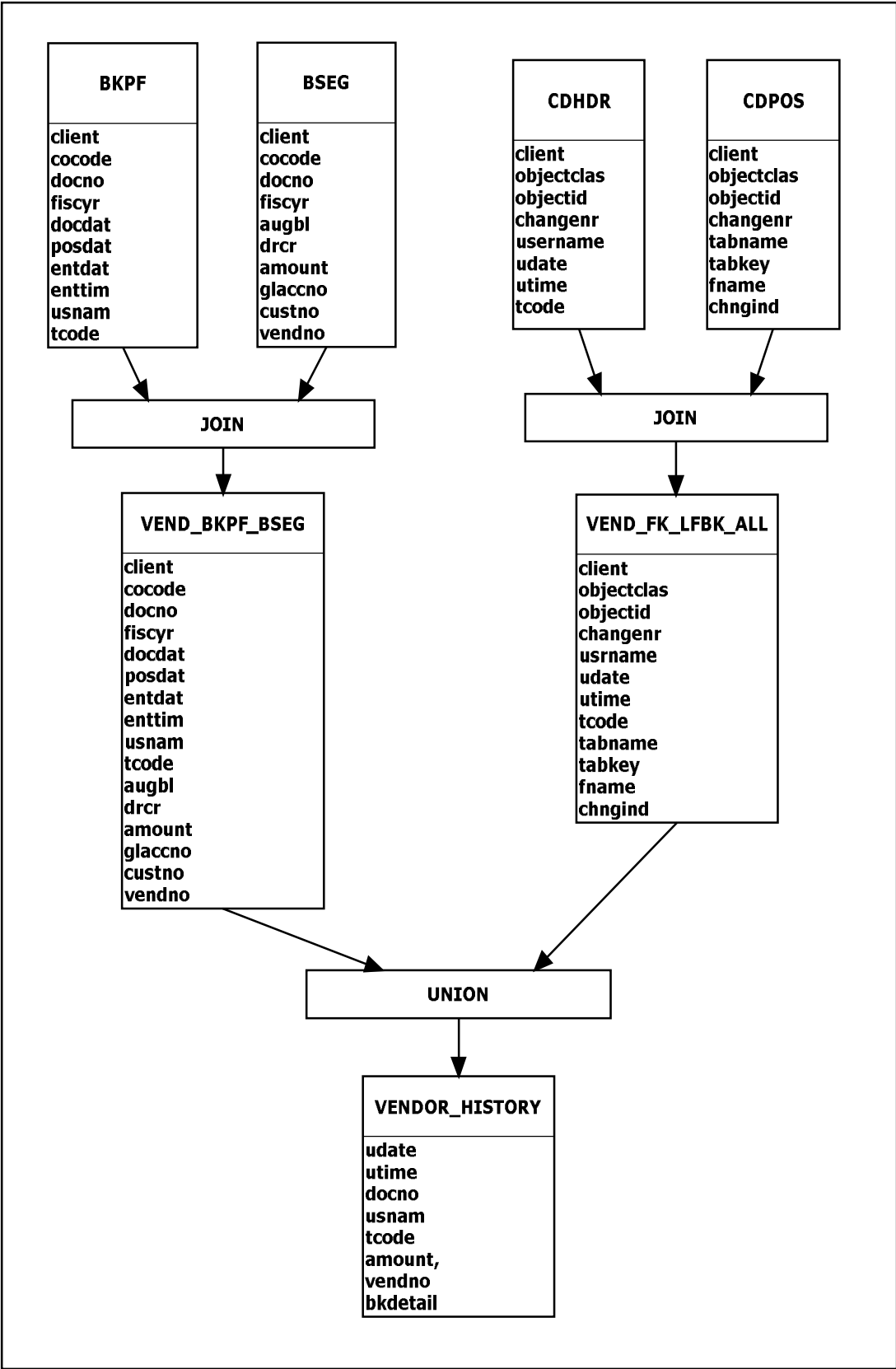



Figure 4.7: Detection – flipping vendor bank account

Fraud symptom 2: Duplicate transactions (Figure 4.8)

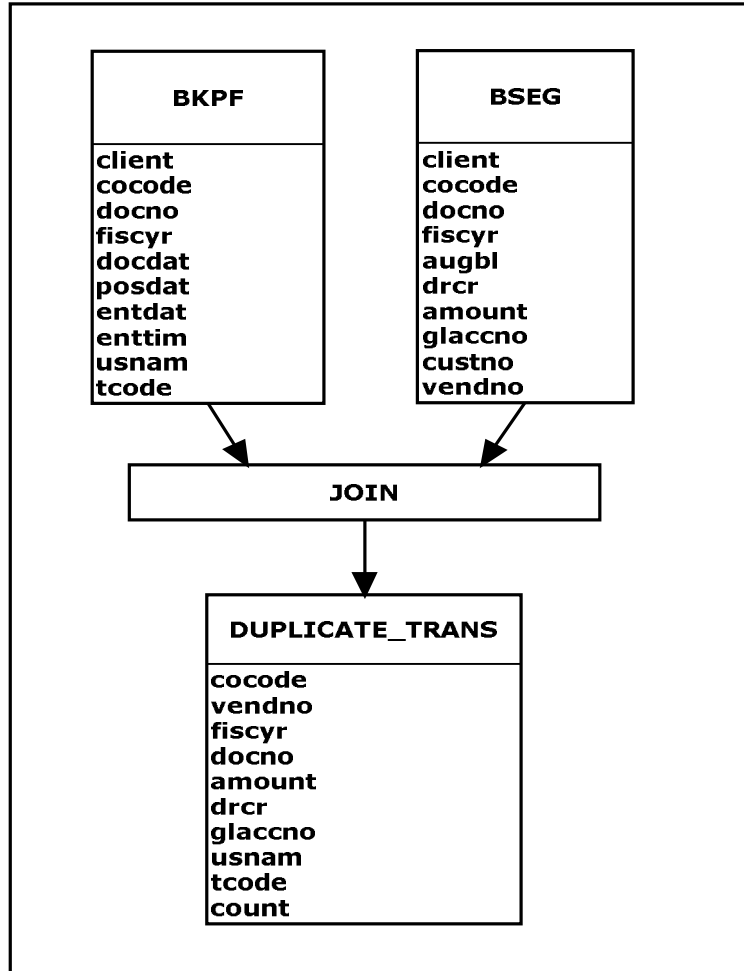


Figure 4.8: Detection – duplicate transactions

```
CREATE TABLE DUPLICATE_TRANS AS
  SELECT cocode, vendno, fiscyr, docno, amount, drcr,
         glaccno, usnam, tcode, count (*) as count
  FROM BSEG JOIN BKPF
  ON bseg.docno = bkpf.docno
  AND bseg.fiscyr = bkpf.fiscyr
  AND bseg.cocode = bkpf.cocode
  GROUP BY cocode, vendno, fiscyr, docno, amount, drcr,
```

glaccno, usnam, tcode
HAVING *count > 1*

Fraud symptom 3: Invoices with round dollar amounts (Figure 4.9)

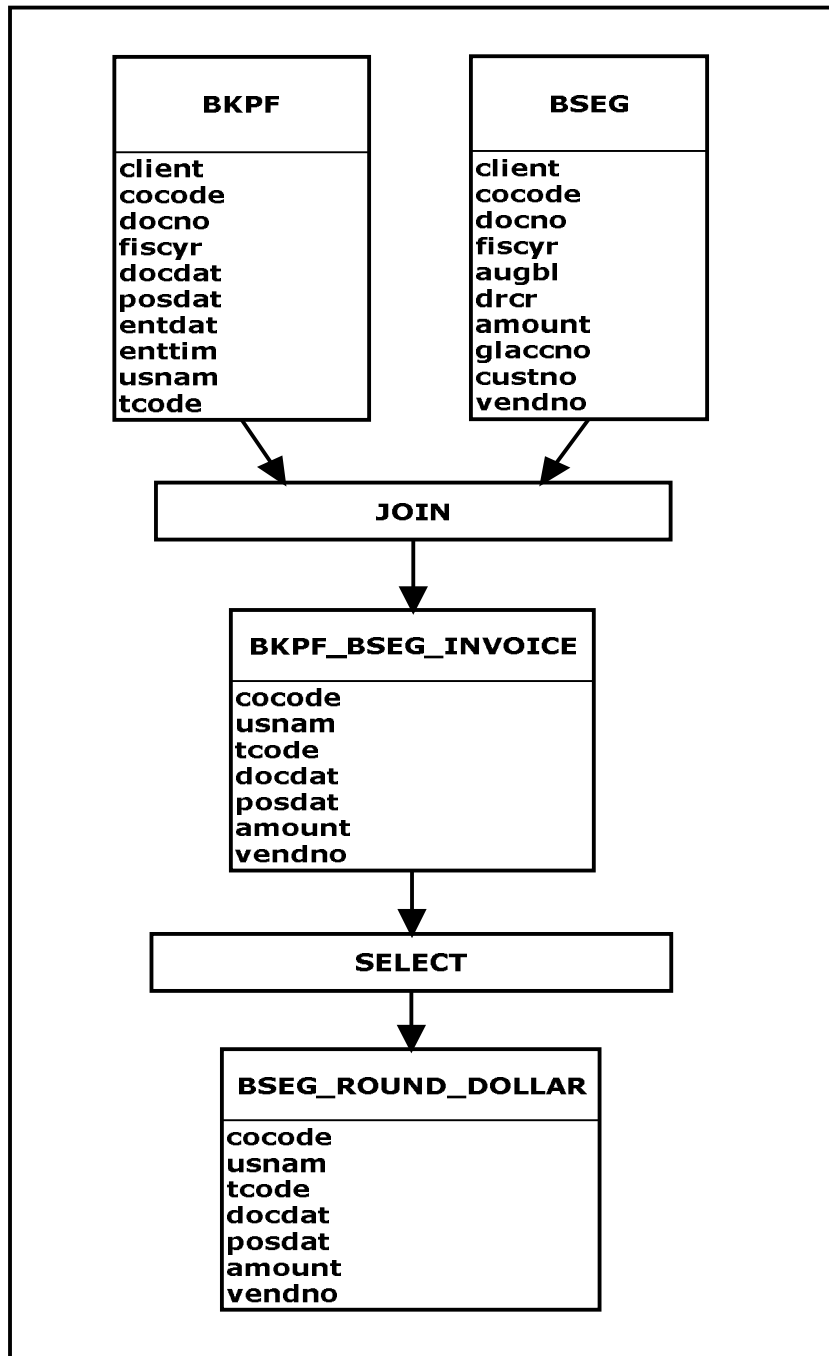


Figure 4.9: Detection – invoices with round dollar amounts

```

CREATE TABLE BKPF_BSEG_INVOICE AS

SELECT cocode, usnam, tcode, docdat, posdat, amount, vendno

FROM BKPF JOIN BSEG

ON bkpf.cocode = bseg.cocode

AND bkpf.docno = bseg.docno

AND bkpf.fiscyr = bseg.fiscyr

WHERE vendno IS NOT NULL

AND tcode IN invoice-tcode-list

```

```

CREATE TABLE BSEG_ROUND_DOLLAR AS

SELECT *

FROM BKPF_BSEG_TRANS

WHERE amount = INT(amount)

```

Fraud symptom 4: Invoices with amounts consistently below approval limit
(Figure 4.10)

```

CREATE TABLE INVOICE_BELOW_APPROVAL

SELECT *

FROM BKPF_BSEG_INVOICE

WHERE (amount <= amounthigh)

AND (amount >= amountlow)

```

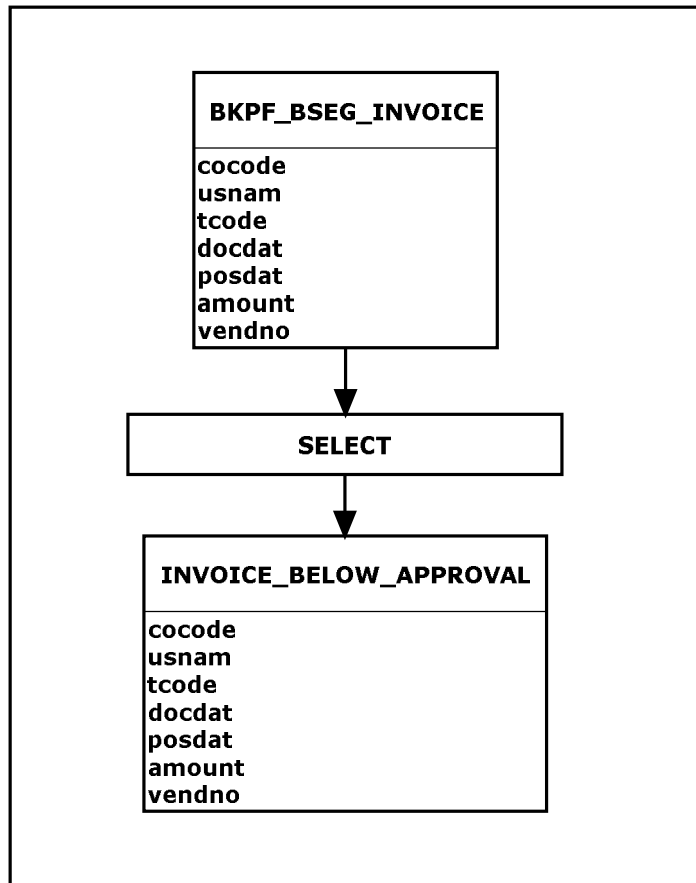


Figure 4.10: Detection – invoices below approval limit

Fraud symptom 5: Vendors with payments that exceed the last largest payment by a significant amount (Figure 4.11)

```

CREATE TABLE BKPF_BSEG_PAYMENT AS

  SELECT cocode, usnam, tcode, docdat, posdat, amount, vendno

  FROM BKPF JOIN BSEG

  ON bcpf.cocode = bseg.cocode

  AND bcpf.docno = bseg.docno

  AND bcpf.fiscyr = bseg.fiscyr
  
```

WHERE *vendno* IS NOT NULL

AND *tcode* IN *PAYMENT-tcode-list*

GROUP BY *vendno*, *amount* DESCENDING

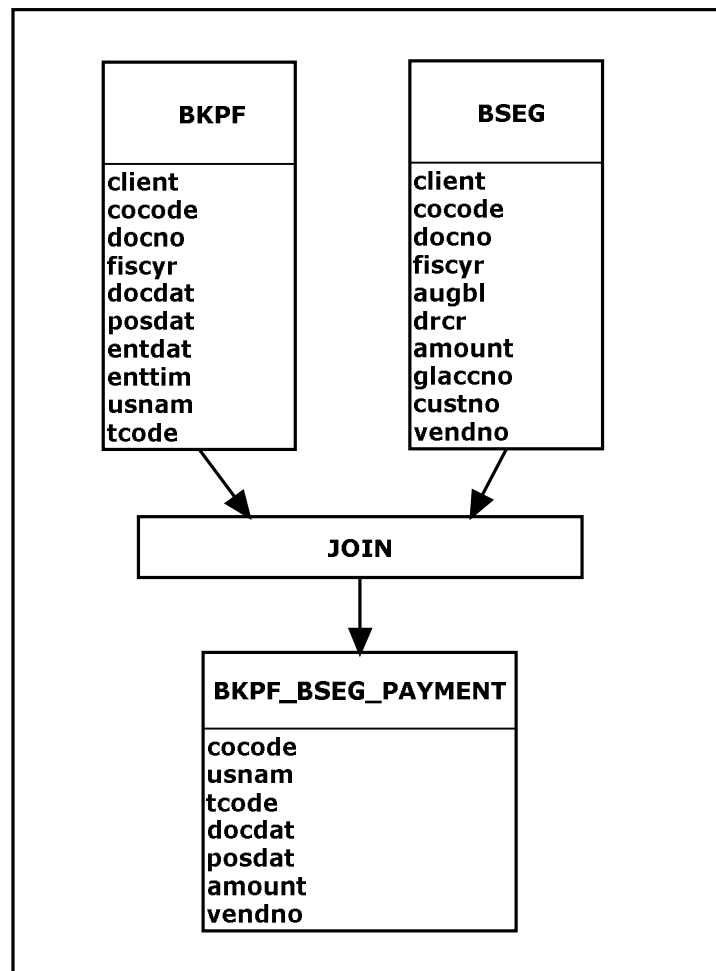


Figure 4.11: Detection – vendor payments exceeding last largest

Fraud symptom 6: Excessive use of one-time vendors (Figure 4.12)

CREATE TABLE *ONETIME_VENDORS*

SELECT *cocode*, *usnam*, *tcode*, *docdat*, *posdat*,

```

amount, vendno, count (*) as count
FROM BKPF_BSEG_INVOICE
WHERE (vendno = onetimevendor)
AND count > threshold

```

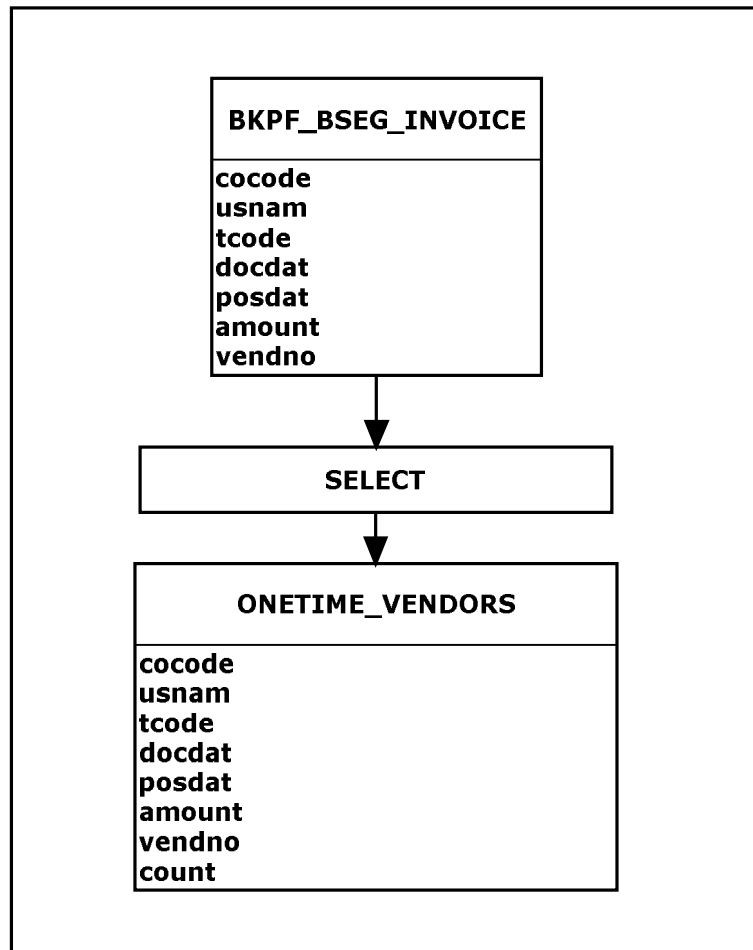


Figure 4.12: Detection – use of one time vendors

Fraud symptom 7: Vendors with similar names (Figure 4.13)

```

CREATE TABLE VENDORS_SIMILAR
SELECT lifnr, name1

```

FROM *LFA1*

WHERE (*name1* = *searchname*)

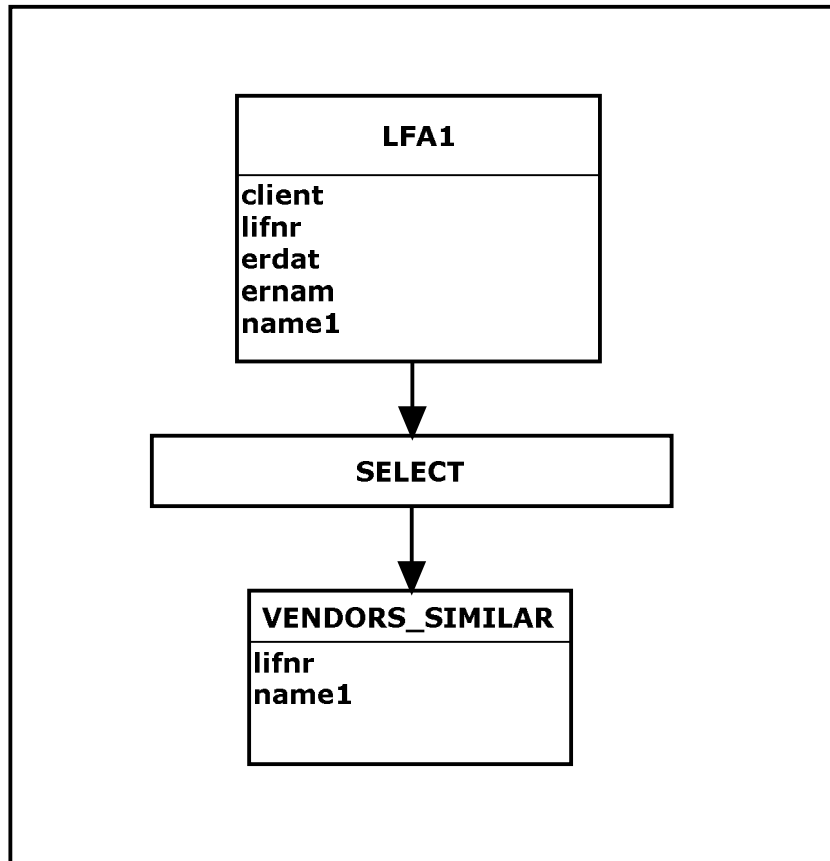


Figure 4.13: Detection – vendors with similar names

Fraud symptom 8: Vendors becoming active after long periods of being dormant (Figure 4.14)

```
CREATE TABLE VENDOR_ACTIVITY AS
```

```
SELECT cocode, usnam, tcode, docdat, posdat, amount, vendno
```

```
FROM BKPF JOIN BSEG
```

```
ON bkpf.cocode = bseg.cocode
```



```

AND bkpf.docno = bseg.docno

AND bkpf.fiscyr = bseg.fiscyr

WHERE vendno IS NOT NULL

AND tcode IN invoice-tcode-list

AND tcode IN payment-tcode-list

GROUP BY vendno, docdat, amount ASCENDING

```

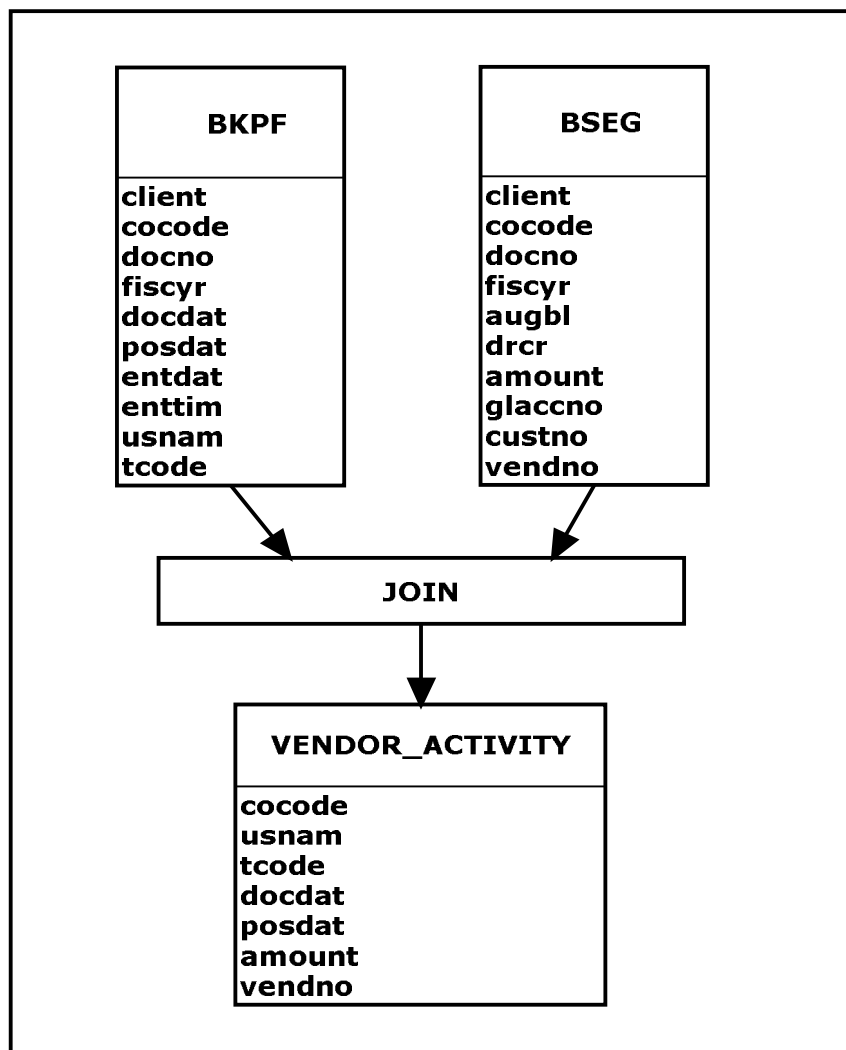


Figure 4.14: Detection – vendors becoming active after long period

Fraud symptom 9: Multiple vendor accounts with different payment details

(Figure 4.15)

```
CREATE TABLE VENDOR_MBANKACC AS
```

```
    SELECT objectid, name1, bkdetail, udate, utime, username  
    FROM CDHDR_CDPOS_LFA1
```

```
CREATE TABLE VENDOR_MULTIBANKACC AS
```

```
    SELECT objectid, count (*) as count  
    FROM VENDOR_MBANKACC  
    GROUP BY objectid  
    HAVING count > 1
```

```
CREATE TABLE VENDOR_MULTIBANK_ALL AS
```

```
    SELECT *  
    FROM VENDOR_MULTIBANKACC  
    WHERE vendor_mbankacc.objectid IN  
        (SELECT objectid FROM VENDOR_MULTIBANKACC)  
    ORDER BY objectid ASCENDING
```

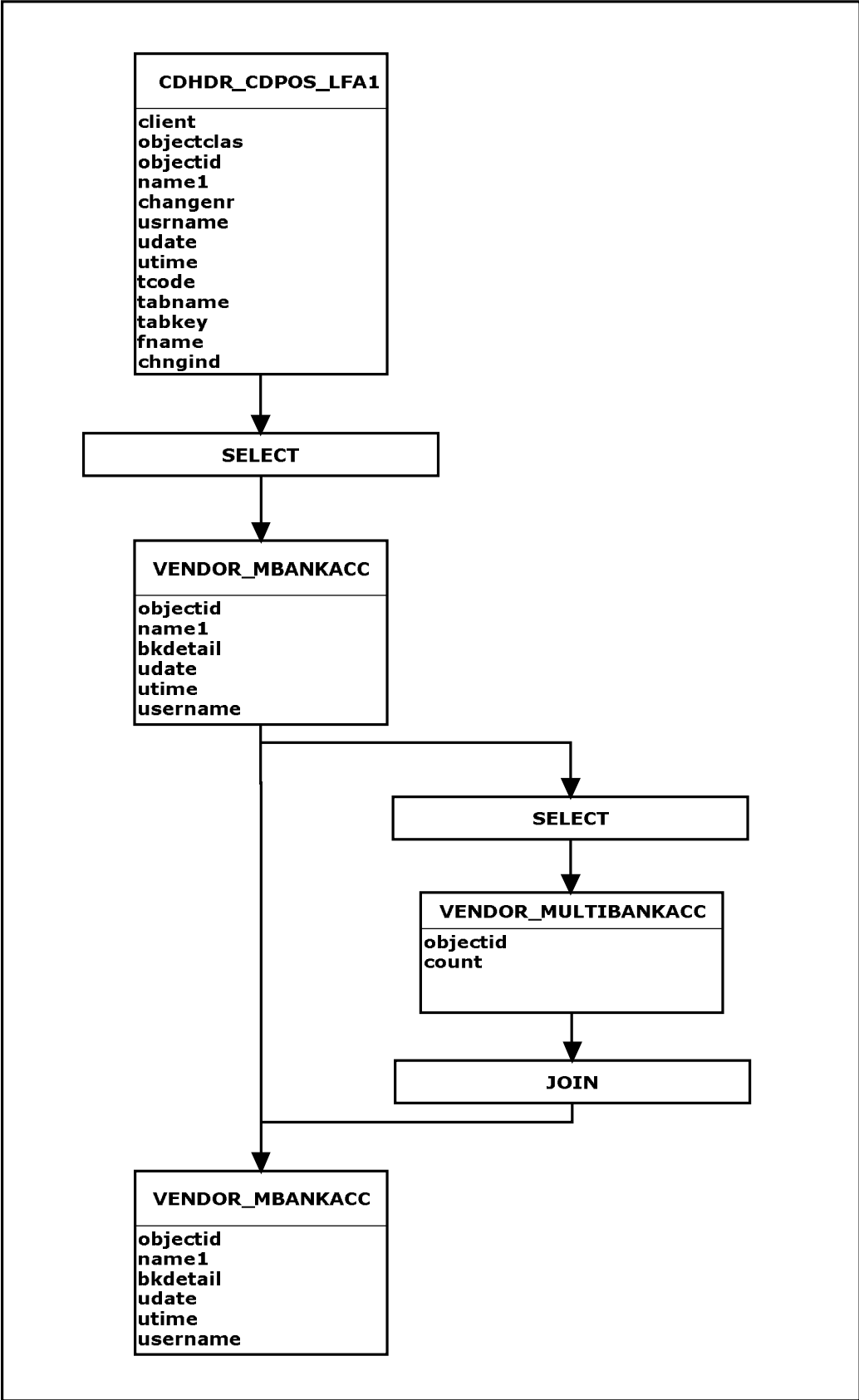


Figure 4.15: Detection – multiple vendors with different payment details

Fraud symptom 10: Multiple vendors sharing the same payment details
(Figure 4.16)

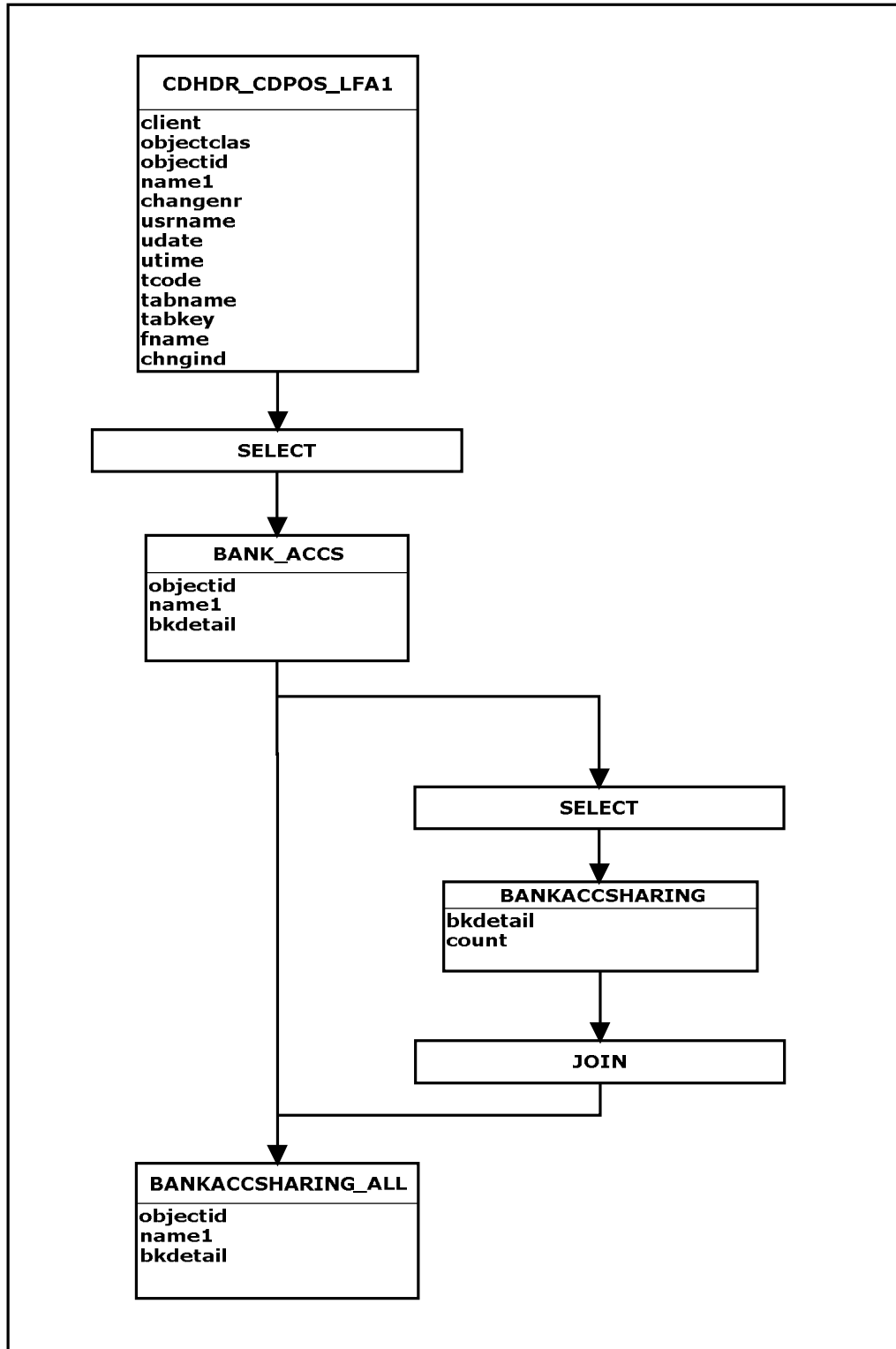


Figure 4.16: Detection – multiple vendors sharing payment details

```
CREATE TABLE BANK_ACCS AS
    SELECT bkdetail, objectid, name1
    FROM CDHDR_CDPOS_LFA1
    ORDER BY bkdetail
```

```
CREATE TABLE BANKACCSHARING AS
    SELECT bkdetail, count (*) as count
    FROM BANKACCS
    GROUP BY bkdetail
    HAVING count > 1
```

```
CREATE TABLE BANKACCSHARING_ALL AS
    SELECT *
    FROM BANKACCS
    WHERE bankaccs.bkdetail IN
        (SELECT bkdetail FROM bankaccsharing)
    ORDER BY bkdetail ASCENDING
```

Fraud symptom 11: Benford's Law analysis of invoices (Figure 4.17)

Benford's law gives expected frequencies of digits in numerical data. Benford found that contrary to belief, digits in tabulated data are not equally likely and are biased towards lower digits. The basic digits tests are tests of the first digits, second digit and first-two digits. These are called the first-order tests. The first digit test is a high-level test of reasonableness that is actually too high-level to be of much use. For

accounts payable and other data sets involving prices, the first-two digits test is a more focused test that detects abnormal duplications of digits and possible biases in the data (Nigrini 2011). The system performs a basic digits test of the first-two digits. Spikes may be indicative of fraud and require further investigation.

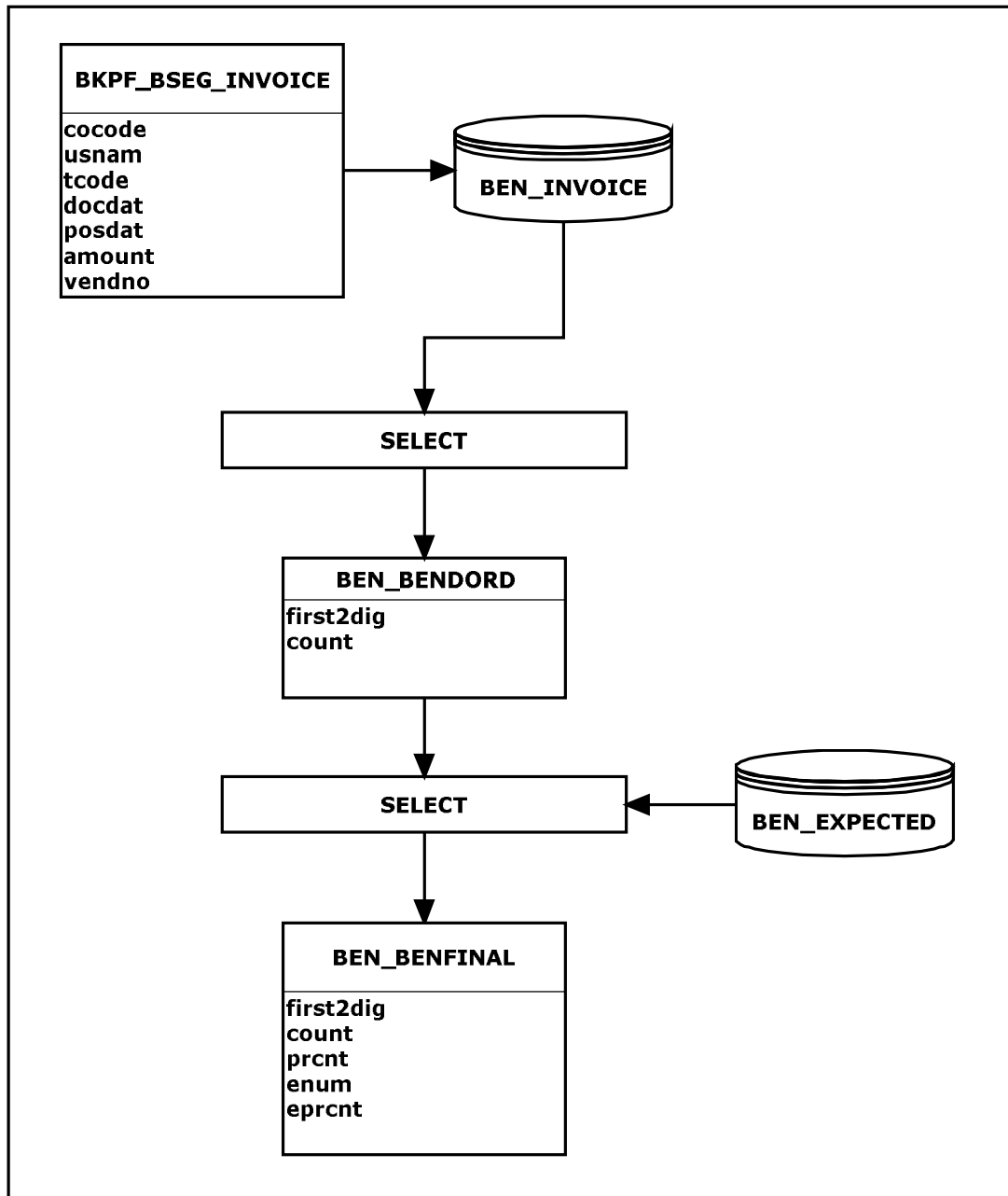


Figure 4.17: Detection – Benford's Law analysis of invoices

DATA *BEN_INVOICE*

SET *BKPF_BSEG_INVOICE*

wrbtrstr = INPUT (*amount*, \$12.)

IF *amount* > 0

frstloc = INDEXC (*wrbtrstr*, '0123456789')

frstdig = SUBSTR (*wrbtrstr*, *frstloc*, 1)

CREATE TABLE *BEN_BENFORD* AS

SELECT *frstdig*, count (*) as *cnt*

FROM *BEN_INVOICE*

GROUP BY *frstdig*

SELECT SUM (*cnt*) INTO: *n*

FROM *BEN_BENFORD*

DATA *BEN_EXPECTED*

INPUT @1 *edig* \$1

@3 *eprcnt* 5.3

enum = INT (*eprcnt* * &*n*)

1 0.301

2 0.176

3 0.125

4 0.097

5 0.079

6 0.067

7 0.058

8 0.052

9 0.046

```

CREATE TABLE BEN_FINAL AS

SELECT frstdig, cnt, prcnt, enum, eprcnt

FROM BEN_BENFORD, BEN-EXPECTED

WHERE frstdig = edig

```

Fraud Risk Index

The *Fraud Risk Index* determines the propensity for fraud occurring in an organisation being investigated. It is calculated from the variables shown in Table 4.6 (this definition is purely for demonstration, further research is recommended in Section 6.4.1 to obtain evidence to adjust values of weights and potentially improve accuracy of the Index).

<p>FRAUD RISK INDEX = riskyusers + vendsharingbank + vendumtbank + vendbankchanges + beninv + benpmt + rinv + rpmt</p>

The calculated value is out of **10**. For example:

$$\mathbf{FRAUD\ RISK\ INDEX = 7 / 10}$$

The *Fraud Risk Index* is intended to create awareness for the potential of fraud occurring in an organisation. A high value does not necessarily indicate that fraud

has occurred or vice versa. Individual circumstances within an organisation must be taken into account when basing decisions on the value of the index.

Table 4.6: Risk Index variables

Input Variable	Range	Output Variable	Value
Risky users	>= 5	riskyusers	2.0
	>= 1		1.0
Vendors sharing bank accounts	>= 100	vendsharingbank	2.0
	>= 50		1.5
	>=20		1.0
	>=10		0.5
	>=1		0.1
Vendors with multiple bank accounts	>= 100	vendmultibank	2.0
	>= 50		1.5
	>=20		1.0
	>=10		0.5
	>=1		0.1
Vendors with multiple changes to bank accounts	>= 100	vendbankchanges	2.0
	>= 50		1.5
	>=20		1.0
	>=10		0.5
	>=1		0.1
Benford's invoice deviations	>= 50	Beninv	0.5
	>= 25		0.4
	>= 10		0.25
	>= 1		0.1
Benford's payment deviations	>= 50	Benpmt	0.5
	>= 25		0.4
	>= 10		0.25
	>= 1		0.1
Value of invoices by risky users	>= 100,000	Rinv	1.0
	>= 50,000		0.75
	>= 20,000		0.50
	>=5,000		0.25
Value of payments by risky users	>= 100,000	Rpmt	1.0
	>= 50,000		0.75
	>= 20,000		0.50
	>=5,000		0.25

4.6. Storage

Audit trail data is required to detect potential fraud in SAP enterprise systems. Additionally, historical data is required to detect frauds that are perpetrated over a period of several days, weeks, or months, for example, flipping of a vendor's bank details, with payments being processed in the interim period (fraud symptom 1, Figure 4.7). Furthermore, the detection engine creates several tables during pre-processing. Data storage is therefore an essential element of the prototype (Figure 4.18).

A data warehouse provides storage for extracted SAP data tables (CDHDR, CDPOS, BKPF, BSEG and LFA1). The system accumulates data for users and vendors and produces reports and visualisations. Tables created during pre-processing are temporarily stored, and continually reused by the detection engine. Imported data is retained for a period of one year in the data warehouse.

Data structure of SAP tables is shown in Figures 4.4 to 4.9 and in Table A8.1. This structure is duplicated in the prototype. Relationships between tables are preserved.

The following relationships exist.

- BKPF and BSEG: *client, cocode, docno, fiscyr*
- CDHDR and CDPOS: *client, cocode, objectclas, objectid, changenr*
- BSEG and LFA1: *vendno*
- BSEG and SKAT: *client, glacno*

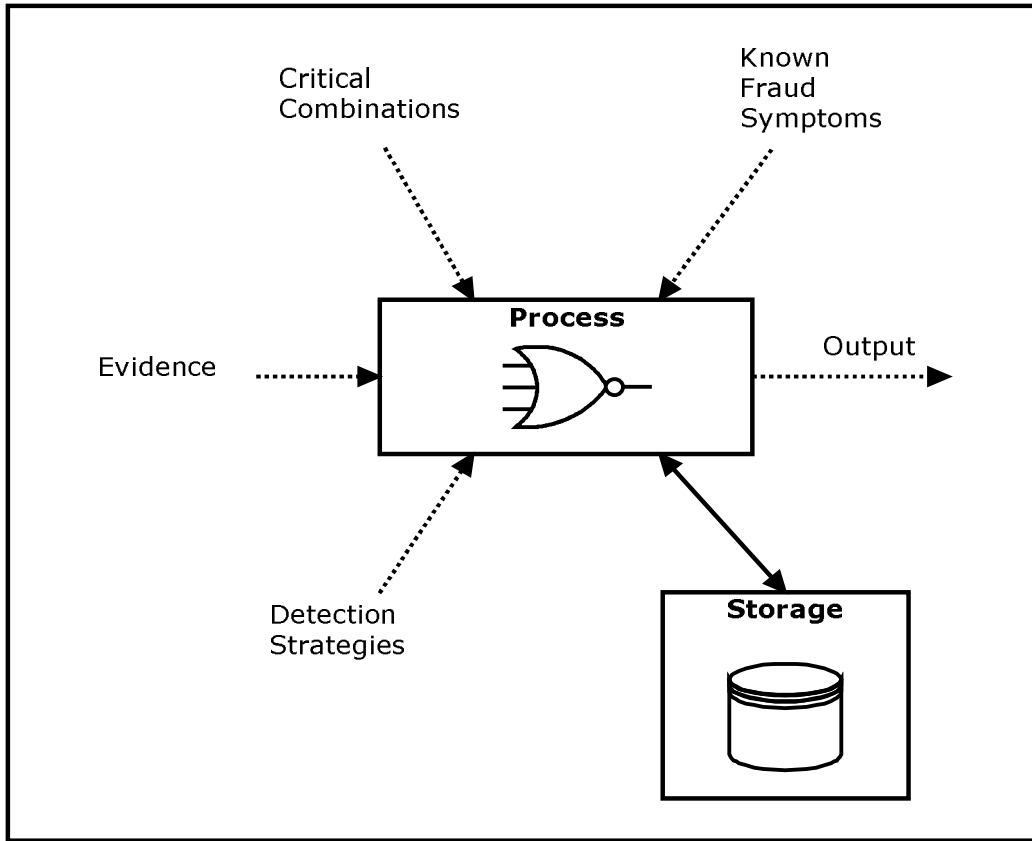


Figure 4.18: Storage module

4.7. Output

The output module (Figure 4.19) provides summarised and detailed reports and visualisations (graphs, charts and diagrams) of user and vendor activities.

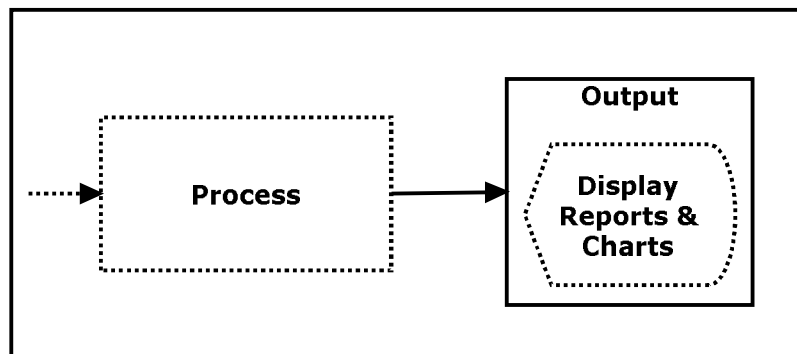


Figure 4.19: Output module

Enterprise systems generate hundreds of thousands to millions of transactions annually. While most of these are legal and routine transactions, a small number may be fraudulent. The enormous amount of generated transactions makes it difficult to find these few instances among legitimate transactions. For a large organisation, this means monitoring hundreds of thousands of transactions and then investigating suspicious ones in depth at considerable expense. The problem becomes overwhelming and is growing worse (Chang et al. 2007).

Visualisation is a general term used to describe any technology that enable users to 'see' information in order to help them better understand and put it in an appropriate context (TechTarget 2010 ; GraphViz 2010). Visualisation tools go beyond the standard charts and graphs, displaying data in more sophisticated ways such as dials and gauges, heat maps, tree maps and detailed bar and pie charts. Patterns, trends and correlations that might go undetected in text-based data can be exposed and recognised easier with visualisation. Visualisation replaces blind querying and extended analysis of transaction data, with contextual analysis. The strength of visualisation lies in its capacity for discovery and the recognition of new insights that are unexpected by users. Empirical evidence confirms that using visualisations results in improved user performance in information seeking tasks (Fetaji 2011). It can therefore be concluded that using visualisations enable an auditor to effectively and efficiently identify anomalous activities in transaction data without the burden of 'information overload'.

Visualised data is frequently displayed in dashboards (Figure A5.1). Dashboards provide users with high-level views of corporate information, metrics and key performance indicators. The images may include interactive capabilities, enabling users to manipulate them or drill into the data for querying and analysis. Indicators designed to alert users when data has been updated or predefined conditions occur, can also be included (Selby 2009).

Output produced by the prototype includes a combination of reports and visualisations. User activity reports include:

- user profiles; activities performed by all system users (Table 4.7, Figures 4.20 and 4.21);
- critical combinations; users that violate segregation of duties; and
- individual user analysis reports; activities performed by an individual being investigated.

Vendor reports include:

- Benford's Law analysis; determines conformity of numbers;
- vendors with multiple bank accounts;
- vendors sharing bank accounts;
- vendors with similar names; and
- individual vendor analysis reports: transaction analysis for an individual vendor being investigated (Figure 4.22).

Table 4.7: Activity summary

T-Code	Transaction Name	Activity
FB60	Enter Incoming Invoices	2305
F-53	Post Outgoing Payments	2135
FK02	Change Vendor (Accounting)	252

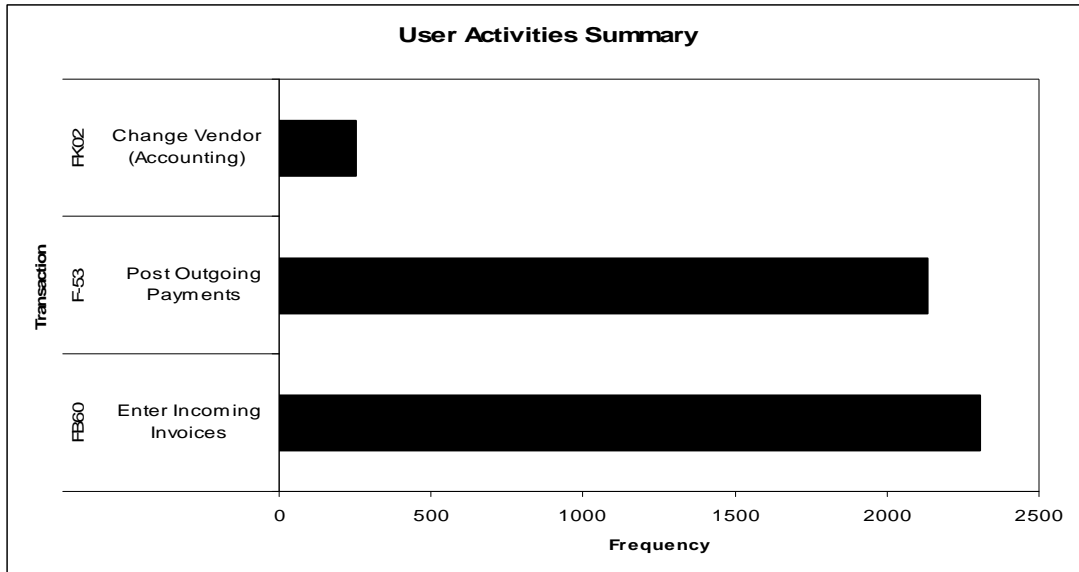


Figure 4.20: Visualisation - activity summary

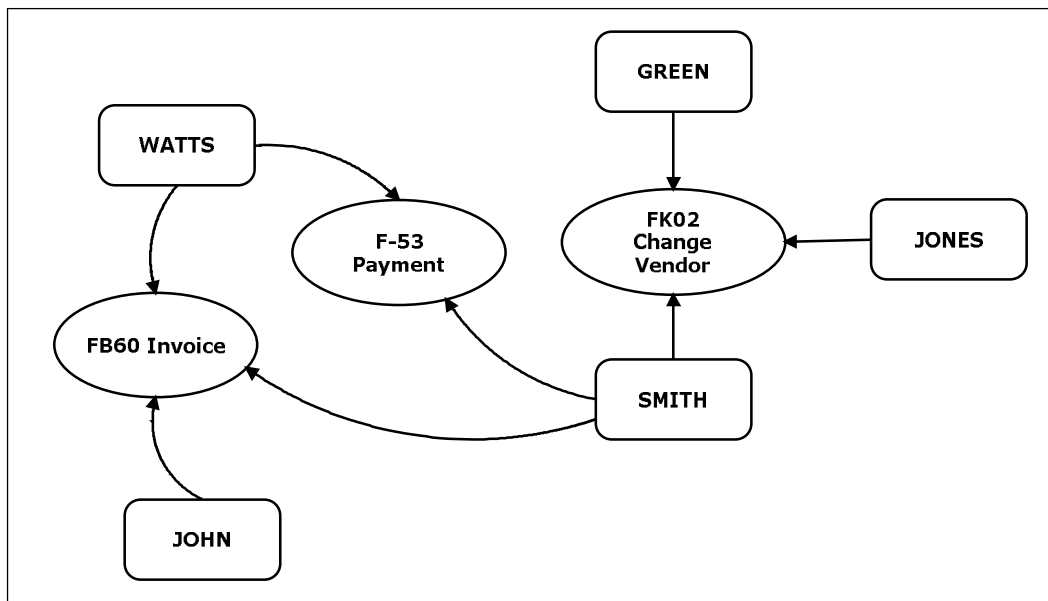


Figure 4.21: Visualisation - user profile

These reports and visualisations are accessed via a web browser interface. The web browser provides a simple, intuitive and graphical user interface. An auditor may use these reports to initiate further investigation of anomalous activities. The intention is to promptly identify potentially fraudulent activities and report these without overwhelming an auditor with excessive information.

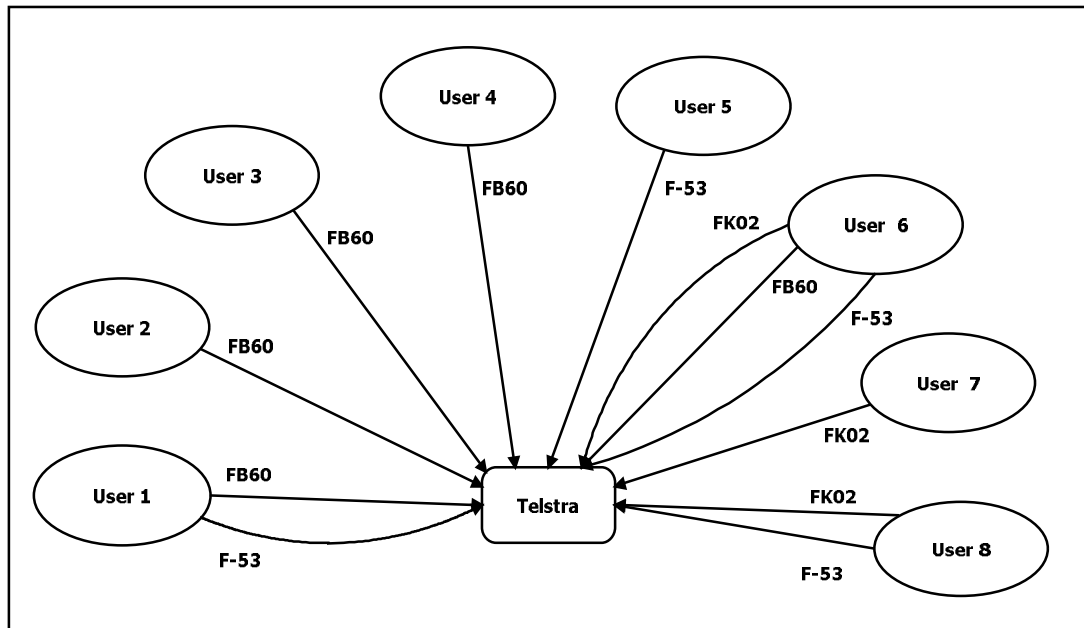


Figure 4.22: Visualisation – interaction between users and individual vendor

4.8. User interface

The user interface is the space where interaction occurs between an auditor and the prototype. The goal of interaction is effective operation and control of the prototype, and feedback from the prototype which aids an auditor in making decisions. Web-based user interfaces accept input and provide output by generating web pages which are viewed using a web browser program. Usability is a main characteristic of the

interface as it ensures that the prototype can be used for its intended purpose by its target audience efficiently and effectively.

Contents	Main Display Area
<ol style="list-style-type: none"> 1. AP Summary <ul style="list-style-type: none"> • Dashboard 2. User Profiles <ul style="list-style-type: none"> • Summary • Vendor Maintenance • Invoices • Payments • All Combinations 3. Critical Combinations <ul style="list-style-type: none"> • Invoice & Payments • Vendor Maint. & Invoice • Vendor Maint & Payments • All Combinations 4. User Activity Analysis <ul style="list-style-type: none"> • Risky Users • Analyze Users 5. Vendor Analysis <ul style="list-style-type: none"> • Analyze Vendors • Benford's Law • Sharing Payment Details • Multiple Bank Accounts • Similar Names 6. Configure System <ul style="list-style-type: none"> • Setup Options 7. HOME 	<p>THIS AREA DISPLAYS</p> <p>SELECTED</p> <p>REPORT</p> <p>OR</p> <p>VISUALISATION</p>

Figure 4.23: User-interface

The interface for the prototype is web-based. It comprises two frames: i) a hyperlinked contents frame; and ii) a main display frame for viewing reports and visualisations (Figure 4.23). The main goal of the web-interface is to ensure that the prototype is easy to learn and use, is user friendly, and provides adequate on-screen instructions and feedback.

4.9. Errors

The purpose of the prototype is to demonstrate the concept of automating fraud detection in enterprise systems. With any scientific or real-world process, there is no such ideal as total proof or total rejection. It is therefore necessary to work with probabilities. This means that, whatever level of detection is reached, there is still the possibility that results may be wrong. Errors in detection fall into two categories, Type I and Type II.

A Type I error, also known as a false positive, occurs when a test rejects a true null hypothesis or general position (Shuttleworth 2008). For example, if the null hypothesis states that round dollar invoices are a symptom of fraud, and round dollar invoices do indeed exist, but the prototype rejects this hypothesis, it may falsely ignore potentially fraudulent transactions.

A Type II error, also known as a false negative, occurs when a test fails to reject a false null hypothesis or general position (Shuttleworth 2008). For example, if the null hypothesis states that round dollar invoices are a symptom of fraud, and there are no round dollar invoices present, but the prototype rejects this hypothesis, it may falsely identify transactions as potentially fraudulent.

A false positive results in legitimate transactions being classified as fraudulent. A false negative results in fraudulent transactions being classified as legitimate. Incorrect detection may occur due to several factors including; poor or lack of segregation of duties within an organisation's enterprise system; collusion between employees to circumvent segregation of duties; new techniques to perpetrate fraud;

incorrectly defined fraud symptoms; inadequate transaction data; or errors in implementation of the prototype.

It is crucial to recognise that the prototype is intended to assist an auditor by facilitating early detection of potentially fraudulent activities. The onus is then on the auditor to further investigate these anomalous activities.

4.10. Verification and validation of prototype

Software verification and validation is the process of checking that a software system meets specifications and that it fulfils its intended purpose (Figure 4.24). It is a disciplined approach to assessing software products that strives to ensure that quality is built into the software and that it satisfies user requirements (IEEE 2004 ; Wallace et al. 1996).

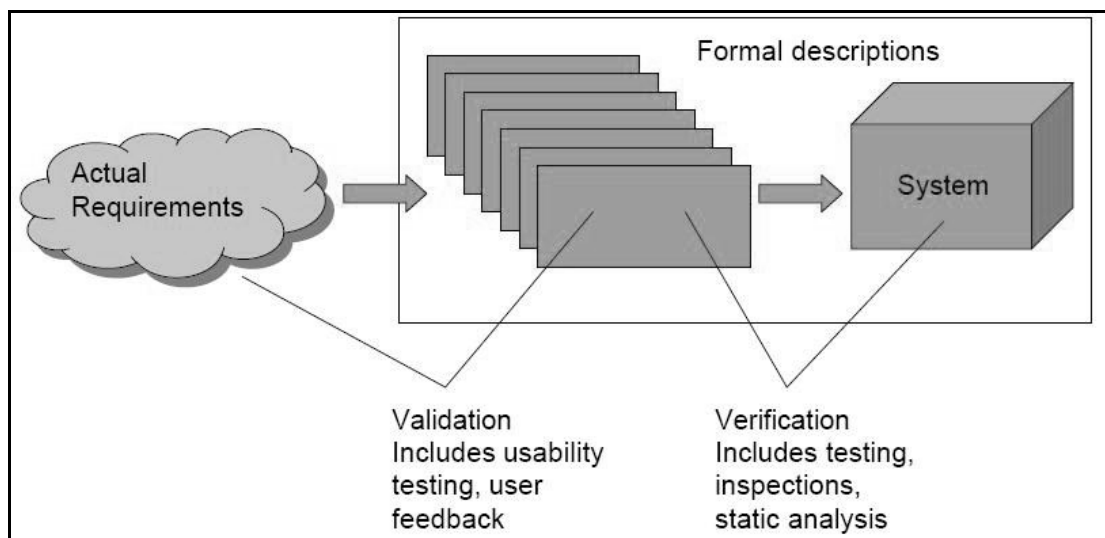


Figure 4.24: Verification and validation

Source: adapted from (Adrion et al. 1982 ; IEEE 2004)

Verification is an attempt to ensure that the product is built correctly and that the outputs of activities meet specifications imposed on them during the design phase.

Software verification looks for consistency, completeness, and correctness of the software and its supporting documentation. Software testing is one of many verification activities intended to confirm that software development output meets its input requirements. Other verification activities may include code and document inspections, walkthroughs, and other techniques (USDoHHS 1997).

Validation is an attempt to ensure that the right product is built and that the product fulfils its specific intended purpose. Validation therefore is the confirmation by examination and provision of objective evidence that software specifications conform to user needs and intended uses, and that the particular requirements implemented through software can be consistently fulfilled. Validation includes useability testing and user feedback.

The prototype as Expert System intended to support a human expert in the decision making process. It is based on computational rules and a knowledge base. The power of the prototype is in the effectiveness and quality of the knowledge it contains. To ensure quality, the knowledge base needs to be verified. Potential problems can be grouped into (Cojocariu et al. 2005):

- consistency problems – caused by unnecessary conditions, redundant or conflicting rules; and
- completeness problems – caused by missing rules, errors, or gaps in the inference chains.

The knowledge base may be logically correct without being valid. Validation measures how well the prototype conforms to what is being modelled. Possible measures may include (Cojocariu et al. 2005):

- productivity measures – to evaluate impact on decisions
- process measures – to evaluate impact on decision making
- perception measures – to evaluate impact on decision makers; and
- product measures – to evaluate technical merits of the prototype.

Verification of the prototype is achieved by performing a series of tests using simulated test data involving simulated activity over a period of one month. Tests include:

- | | |
|-------------------------|--|
| User profiles: | Users are profiled to determine the scope of activities they have performed. Activities include vendor maintenance, invoicing and payment transactions. Summary and detailed reports are produced. |
| Critical combinations: | Users that violate segregation of duties are identified and a report of potentially risky users is produced. |
| User activity analysis: | An individual user is identified from the risky users list and selected for detailed investigation. Reports documenting individual user activities are produced. |
| Vendor analysis: | A series of investigations are performed on active vendors, including vendors sharing bank accounts, vendors with multiple bank accounts, vendors with multiple master records, and Benford's law. |

These tests serve to assess whether the software performs correctly, that it meets the specifications imposed on it, and to provide a demonstration of the potential use of the prototype.

Validation of the prototype is achieved by obtaining independent reviews from auditing practitioners. The reviewer(s) are provided with a summary paper (Singh et al. 2011), a one-hour presentation and demonstration of the prototype. The demonstration involves processing and analysing of actual transaction data. Feedback is requested on the following issues.

1. The importance of such a project for auditing in your organisation.
2. The role that automated fraud detection software could play as an auditing tool for internal auditors.
3. The desirability of a retrospective analysis software tool implemented on a standalone computer system as compared with a system embedded within an enterprise system.
4. The functionality of the prototype, in particular the user interface, reporting and graphical features.
5. Any further comments or suggested improvements to the prototype.

Feedback is also obtained from a panel of experts from CPA Australia - Queensland Division (IT Discussion Group) and ISACA⁴, Queensland Chapter. A short presentation and demonstration is made to panel members (20 minutes). Members are provided an opportunity for a hands-on session using the prototype. Their

⁴ Information Systems Audit and Control Association

feedback is sought using a survey (Appendix 3) on the following key issues, namely *operation, reporting and visualisations, accuracy & efficiency, and impact on auditor productivity.*

Verification is a formal or informal argument that the prototype works on all possible inputs and validation is a process designed to increase our confidence that the prototype works as intended (Cojocariu et al. 2005). A series of tests and feedback from experts assesses the prototypes suitability as a proactive fraud detection tool.

4.11. Prototype design and propositions addressed

This Chapter addresses the following three research propositions.

RP1a: Enterprise system audit trails document adequate data to allow retrospective monitoring of user activities.

SAP audit trails are examined. It is established that they provide detailed descriptions of functions performed by users within an enterprise system. Each function has a transaction code associated with it (for example, FB60 – Enter Vendor Invoice). Each transaction code executed by a user is recorded in the audit trail (Best 2000). Audit trail data is stored in several tables within the SAP enterprise system (Figure 4.3). This data documents changes to master records and accounting audit trails. Thus, it is established that enterprise system audit trails document adequate data to allow retrospective monitoring of user activities. RP1a is therefore supported.

RP1b: Violations in segregation of duties can be identified by analysing audit trails for critical combinations of user activities.

This study supports the principles of segregation of duties within accounts payable (AP) as proposed by Little and Best (2003) and discussed in Chapter 3. Evidence supporting these principles is obtained by examining data from SAP audit trails. It was determined that this data allows association of actions with users'. Critical combinations of user activities have been designed in section 4.4.1. Table 4.3 and 4.4 lists the combination of activities a user has to perform in order to violate each of the SoDs principles. If any of these violations are identified then further investigation of an offending user's activities is necessary to determine whether any fraudulent transactions have been performed. Therefore, it is feasible to detect violations in segregation of duties with available data. Thus, it is established that violations in segregation of duties can be identified by analysing audit trails for critical combinations of user activities. RP1b is therefore supported.

RP1c: Potentially fraudulent transactions can be identified by investigating user activities that violate segregation of duties, match known fraud symptoms, or appear otherwise anomalous.

Given the ability to identify violations in segregation of duties, it is feasible to detect potentially fraudulent transactions made possible by these violations. For example, the ability to identify users who have changed vendor details, entered an invoice and paid the invoice permits detection of potential accounts payable fraud. In addition,

further potential fraud can be detected through examination of other anomalous activities. A catalogue of known fraud symptoms informs this process (Table 4.5).

Thus, it is established that potentially fraudulent transactions can be identified by investigating user activities that violate segregation of duties, match known fraud symptoms, or appear otherwise anomalous. RP1c is therefore supported.

4.12. Conclusion

This Chapter addresses research propositions RP1a, RP1b and RP1c. The SAP enterprise system was investigated and it was concluded that it documents adequate data in its audit trails to allow retrospective monitoring of user activities. A conceptual design of a prototype is proposed. The objective of the prototype is to analyse audit trail data for critical combinations of user activities and to report these to an auditor. Individual users or vendors may subsequently be selected for further detailed investigation to determine whether fraud has potentially been perpetrated.

The conceptual design of the prototype consists of four interrelated modules i.e. *Input, Process, Storage* and *Output*. Detailed design specifications are produced for each of these modules. A logical design is subsequently developed (Figure 4.25) from the design specifications. A two phase strategy for proactive detection of potential fraud is proposed. In phase one, transaction data is periodically extracted from SAP tables CDHDR, CDPOS, BKPF, BSEG, and LFA1. In phase two, the extracted transaction data are pre-processed and, reports and visualisations are produced.

The design objectives defined in this Chapter are implemented in Chapter 5.

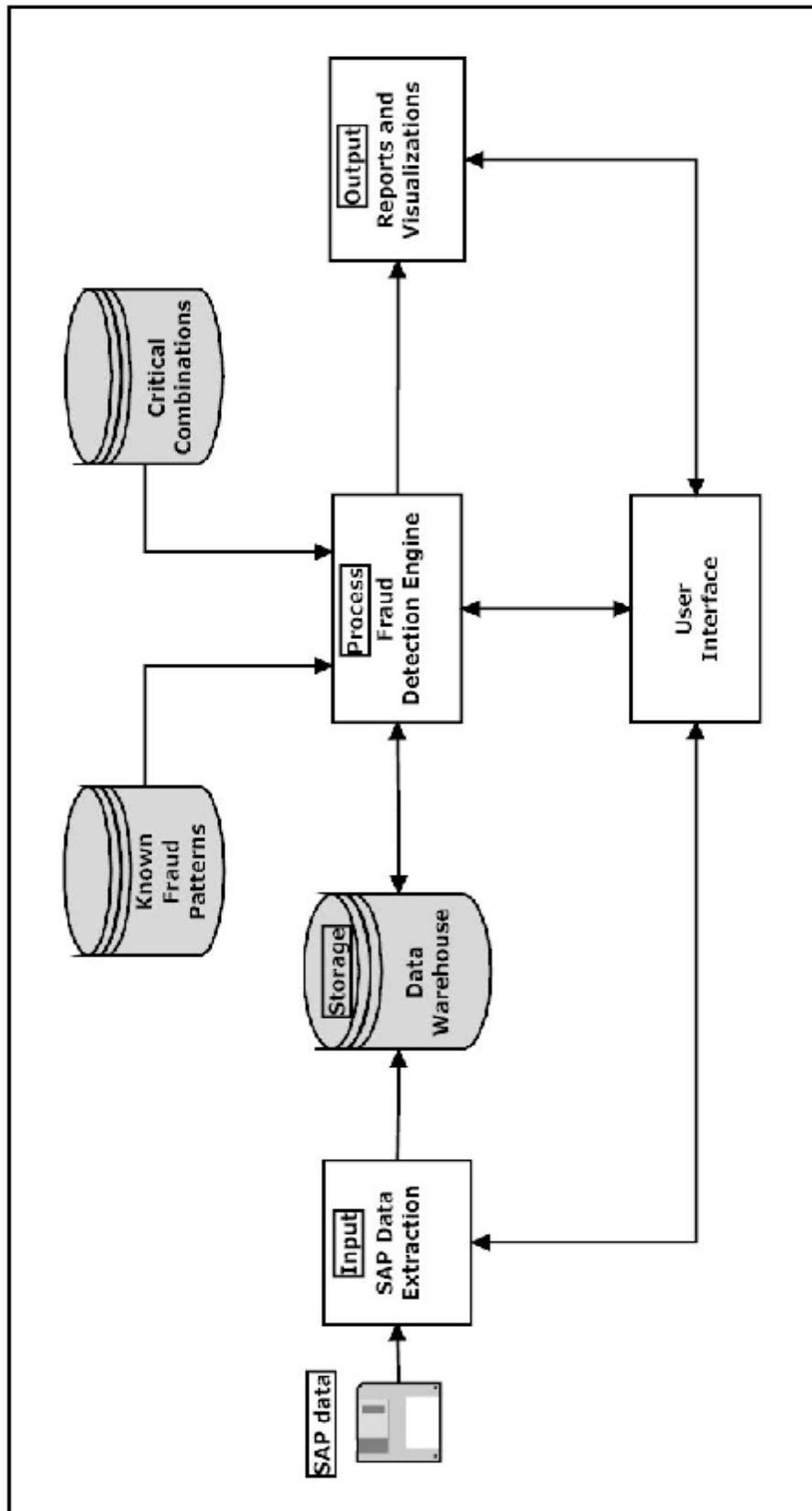


Figure 4.25: Prototype logical design

CHAPTER 5

Prototype Implementation and Testing

5.0. Introduction

Increasing use of information technology has made it essential for auditors to perform their practice using software tools that run on personal computer systems (Kotb and Roberts 2011). This Chapter examines the implementation and testing of the prototype software designed in Chapter 4, for detection of potential fraud. The prototype is implemented as a stand-alone MCL-based software application. The issue of verification and validation of the prototype is addressed. Verification ensures that a product has been built according to the requirements and design specifications. Validation ensures that a product meets the users' needs, and that the specifications were correct in the first place. Verification is addressed by testing the prototype using simulated data. Validation is addressed by obtaining survey feedback from an expert panel and by independent reviews.

This Chapter addresses the following three research propositions.

- § **RP2a:** Software can be developed to identify potentially fraudulent activities and report these using an intuitive visual interface.
- § **RP2b:** Threat monitoring and potential fraud detection can be implemented on a stand-alone external computer system operating independently of an organisation's enterprise system.

§ **RP2c:** Efficiency and effectiveness of the audit process can be improved by using technology to perform continuous monitoring of an organisation's enterprise system.

5.1. Prototype implementation

This section summarises the implementation of the prototype. It describes the environment used for developing the prototype, requirements for execution, data extraction and pre-processing. A process map lists fundamental steps in using the prototype. An analysis process is proposed for a novice system user.

5.1.1. Workstation environment

The prototype has been developed using an Intel Core i5-based computer system with 500 gigabytes hard disk storage and 4 gigabytes of RAM. This is a standard configuration these days and therefore could act as a benchmark for the prototype. The operating system is Windows 7 Home Premium. All results reported in this Chapter are based on experiments performed using this configuration.

5.1.2. Development environment

The primary development environment for the prototype is SAS v9.2 for Windows. The system requires Base SAS and SAS/GRAPH to run. The user interface is web-browser based and requires Microsoft Internet Explorer 5.0 or above. Aspects of the web programs are written in VBScript. Visualisation components are written in GraphViz v1.01.

The purpose of the prototype is to demonstrate the feasibility of the concept of implementing proactive detection of potential fraud in practice, thereby satisfying research propositions RP2a, RP2b and RP2c. A development tool is needed to generate quality programs rapidly having querying, reporting and analytical capabilities. The tool needs the ability to process large datasets. SAS was selected, instead of a specialised language (for example, SQL or Java), because it is a scalable, integrated software environment specifically designed for data access, transformation and reporting. SAS also provides a range of tools for generating visualisations. The user interface, being web-based, provides a simple, intuitive, graphical user interface that most users are already familiar with (Figure 5.1).

The programs used to implement the '*detection engine*' and the '*user interface*' components of the prototype are developed in accordance with the design specifications discussed in Chapter 4. The total disk storage requirements for all of the programs are 3.9 megabytes.

Sample screens produced by the prototype are provided in Appendix 4. The system is GUI-based, requiring an auditor to select (click) an option from the menu. To illustrate, Figure A4.1 displays the Start-up Screen. Clicking '*Configure System*' displays the system configuration options (Figure A4.12)

5.1.3. Data extraction and pre-processing

Data requirements for fraud detection in SAP enterprise systems are discussed in Section 4.3.1. Accounting audit trails are routinely extracted from a SAP system and imported into the prototype for pre-processing and analysis. These audit trails are



Figure 5. 1: User interface

stored in tables **BKPF**, **BSEG**, **CDHDR**, and **CDPOS**. Vendor general data is stored in table **LFA1**. Table A8.1 provides details of tables and fields required for data extraction.

SAP data tables may be accessed through the SAP menu or by using transaction codes (t-codes) as shown below (Table 5.1).

Table 5.1: Source of data to detect known fraud symptoms

T-Code	Menu path
SE11 (ABAP Dictionary)	SAP Menu > Tools > ABAP Workbench > Development > ABAP Dictionary
SE16 (Data Browser)	SAP Menu > Tools > ABAP Workbench > Overview > Data Browser

Table A8.2 summarises the procedure for extraction of the data tables. Each table requires a different filter for extraction. For example, table **BKPF** may be extracted by date range, but there is no such filter for table **BSEG**. Individual extraction requirements are documented in Table A8.3. This document also serves as a record for data extraction.

Extracted data tables are imported, and preformatted before being pre-processed by the prototype (Figure A4.16). No data manipulation occurs during pre-formatting. Pre-formatting is required as different versions of SAP output extracted data in different formats. Pre-formatting ensures that extracted data is imported into the prototype in a standard format. An auditor has the option to: i) create a new data warehouse (on first use); or ii) append data to an existing data warehouse (Figure A4.17). Filter parameters for date range (Figure A4.12) and approval limits for

invoices and payments (these vary among organisations) are also required (Figure A4.14). On completion of pre-processing reports and visualisations are produced (*note: total disk requirements for one year's data storage is approximately one gigabyte – estimate is based on six months data from case study*). Further detailed investigation of user (Figure A4.6) or vendor (Figure A4.9) transactions may occur at the discretion of an auditor.

The process map shown (Figure 5.2) lists nine fundamental steps in using the prototype (last two steps are initiated by an auditor)

- i). Extract SAP tables as text files.
- ii). Copy text files to system folder.
- iii). Pre-process text files.
- iv). Create new data warehouse or append data to existing data warehouse.
- v). Set date range for analysis.
- vi). Set approval limits.
- vii). Prepare reporting data.
- viii). *Perform analyses.*
- ix). *Document findings.*

5.1.4. Reporting system

Several reporting options are available to an auditor. The following analysis process is proposed for a novice system user (Figure 5.3).

Analysis is initiated by examining the dashboard (Figure 5.4). The dashboard provides a high-level overview of various activities performed in an accounts

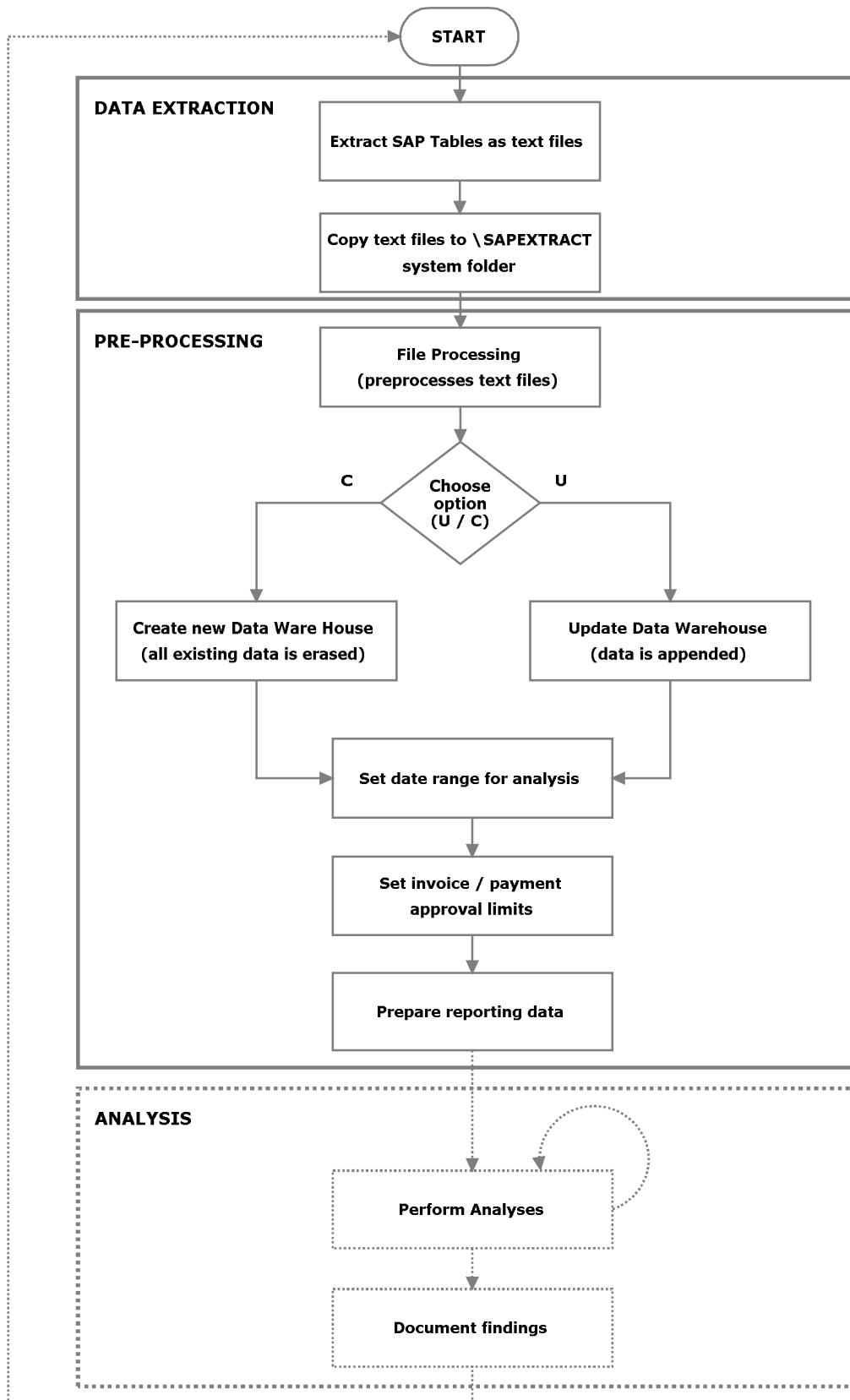


Figure 5.2: Process map – complete

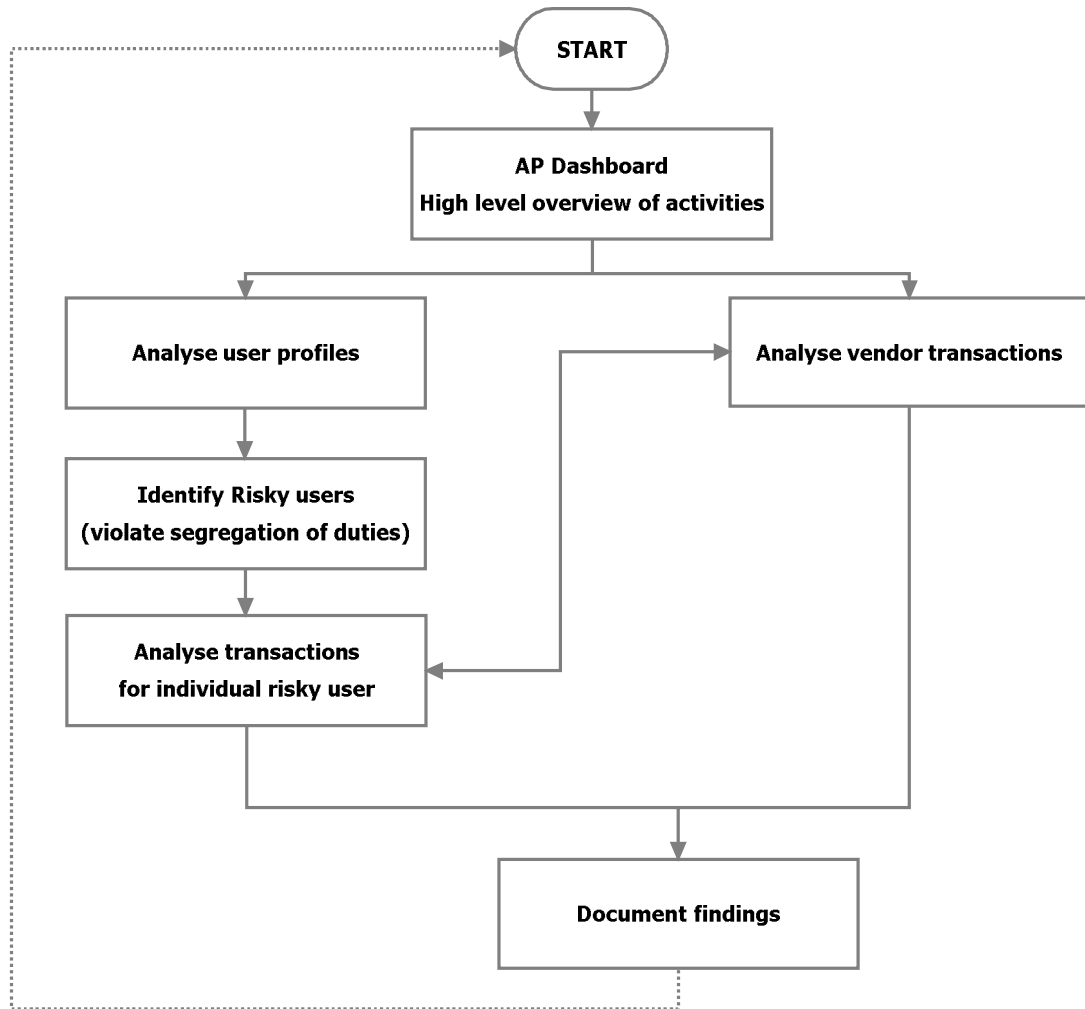


Figure 5.3: Analysis process

payable system. Several indicators on the dashboard are hyperlinked to detailed reports. The *Fraud Risk Index* determines the propensity for fraud occurring in an organisation being investigated. Analysis proceeds by examining user profiles or vendor activities.

User profiles menu (Figure A4.3) provides access to reports documenting overall activities performed by users in an enterprise system. Figure A4.4 provides several reports identifying users that violate segregation of duties principles (discussed in section 4.4.1).

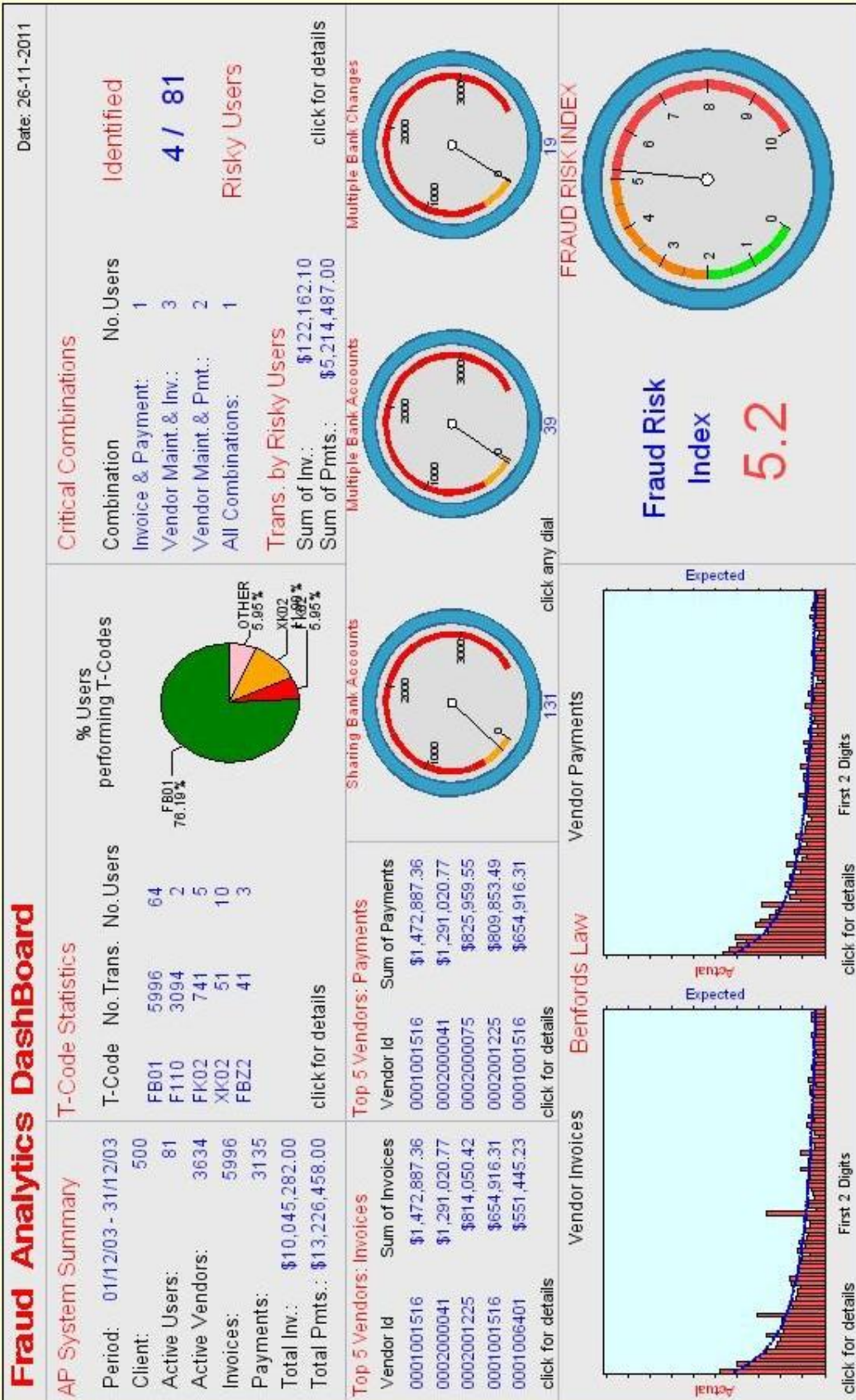


Figure 5.4: Dashboard

A report of risky users (Figure A4.5) provides an auditor with a basis for further investigation of individual users (Figure A4.6). Figure 5.5 shows a list of reports produced dynamically for an individual user. Activity reports for individual users include: i) bank account changes; ii) invoice transactions; iii) payment transactions; iv) duplicate transactions; and v) vendors that individual user has interacted with.

Vendors may be examined to identify anomalous transactions. Figure A4.8 shows a list of reports available to assist an auditor in this process. The following reports are available for investigating vendors: i) Benford's analysis of invoices and payments; ii) vendors with multiple bank accounts; iii) vendors sharing bank accounts; iv) vendors with multiple changes to their banking details; v) top 5 vendors by invoices; and v) top 5 vendors by payments. Vendor names may also be investigated to determine whether any similarities exist (Figure A4.14).

Individual vendors may be selected for further investigation (Figure 5.6). Figure A4.10 shows a list of reports produced dynamically for an individual vendor. Reports for individual vendors include: i) summary transaction statistics; ii) bank account changes; iii) transaction history; iv) duplicate transactions; and v) vendor payments.

Drill-down reporting capabilities enable an auditor to investigate detailed activities for any individual user or vendor. Charts and visualisations enhance the usability of the software and enable rapid identification of patterns and trends of activity without the need to analyse pages of reports, thereby reducing information overload on an auditor.

User Activity Analysis REPORTS

INSTRUCTIONS

1. Click a report link below.
2. Click **Back** in the report screen to return here.

Transaction Reports for this User	Visualizations for this User
<ol style="list-style-type: none">1. Bank Account Changes2. Summary of Invoice Transactions3. Summary of Payment Transactions4. Details of Invoice Transactions5. Details of Payment Transactions6. Round Dollar Invoices7. Round Dollar Payments8. Invoices 5% Below Approval Limit9. Payments 5% Below Approval Limit10. Duplicate Transactions	<ol style="list-style-type: none">1. Vendors Touched by this User2. Analyse interaction between this User and selected Vendor <p>Enter Vendor Id: <input type="text"/></p> <p><input type="button" value="Run SAS Program"/> <input type="button" value="View Report"/></p>

Figure 5.5: User activity reports

Analysis of Vendor Transactions

INSTRUCTIONS

1. Click a report link below.
2. Click **Back** in the report screen to return here.

Transaction Reports for this Vendor	Visualizations for this Vendor
<ol style="list-style-type: none">1. Summary Transaction Statistics2. Bank Account Changes3. Transaction History4. Duplicate Transactions	<ol style="list-style-type: none">1. Analysis of Vendor Payments <i>(to identify Payments that vary by large amounts)</i>

Figure 5.6: Individual vendor reports

The analysis process described above is intended for novice users. Expert users may follow a process of their own design. The user interface is designed with this capability in mind. All options are hyperlinked allowing an auditor to navigate through the interface in no particular order.

In the next section the prototype is verified and tested to determine whether it conforms to the specifications designed in Chapter 4.

5.2. Verification and testing of prototype

Verification attempts to ensure that products are built correctly, and that outputs of activities meet specifications imposed on them. In terms of evaluating software, verification is a process that determines whether products satisfy the conditions imposed during the development phase (IEEE 2004 ; Wallace et al. 1996).

Tests of the prototype were conducted in three phases.

1. **Test data** - the system was tested using a set of **test data** involving simulated activity over a period of one month. This test served to assess whether the software performed correctly, and that it met the specifications imposed in the fraud detection framework developed in Chapters 3 and 4. A detailed trace of the processing of this test data is included in Appendix 5. This trace also acts as a demonstration of the potential use of this software.
2. **Case study 1a** - six months of **actual transaction data** was processed using the prototype. This data was obtained from a large international manufacturing company. These tests exposed the

prototype to live data. This test serves to provide support for research propositions RP2a and RP2b. Data was also collected on processing time and serves as a basis for determining auditor productivity. A detailed trace of the processing of this data is included in Appendix 6.

3. **Case study 1b** - one week of **actual transaction data** from the above organisation was processed using the prototype. This test provides data on processing times and additional support for the research propositions.

Each of the tests is discussed in the following sections.

5.2.1. Test data

Analysis period 01/12/2003 to 31/12/2003 – 1 month

A series of 'manual' experiments were performed on the test data to establish control values for several indicators as described in section 4.10. These experiments were performed using Microsoft Excel. The same experiments were subsequently performed using the prototype and the values produced were reconciled with the control values. Inconsistencies in results were used to correct errors in the prototypes computational rules and knowledge base.

Control values

Control values were produced by manual tests. These are shown in Tables 5.2 to 5.6.

Table 5.2 lists activities performed by users.

Table 5.2: Control values - activities performed by users

# Transactions	Description of Activity	# Users
5,996	posting document (FB01)	64
3,094	parameters for automatic payment (F110)	2
741	change vendor – accounting (FK02)	5
51	change vendor – centrally (XK02)	10
41	post outgoing payments (FBZ2)	3

Table 5.3 lists the number of violations of segregation of duties that have occurred.

Table 5.3: Control values - violation of segregation of duties

# Users	Combination
3	vendor maintenance & invoices
2	vendor maintenance & payments
1	invoices & payments
1	vendor maintenance & invoices & payments

Table 5.4 lists activities performed by user 1USRARSCP. This user was selected for further investigation due to having violated segregation of duties.

Table 5.4: Control values - activities performed by user 1USRARSCP

Activity	#	Total value
Bank account changes	19	n/a
Invoices	7	\$ 1,363.13
Payments	38	\$ 5,214,477.05
Round Dollar Invoices	0	0
Round Dollar Payments	4	\$ 710.00
Invoices 5% below approval limit	0	0
Payments 5% below approval limit	0	0
Duplicate transactions	0	0
Vendors touched	40	n/a

Table 5.5 provides a summary of vendor transactions.

Table 5.5: Control values - summary of vendor transactions

Activity	Value
number of invoices entered	5,996
number payments processed	3,135
total value of invoices entered	\$ 10,045,281.90
total value of payments processed	\$ 13,226,457.57
top vendor by invoice	0001001516
total value of invoices for this vendor	\$ 1,472,887.36
top vendor by payment	0001001516
total value of payments for this vendor	\$ 1,472,887.36
no. active vendors	3,634
no. vendors sharing bank accounts	131
no. vendors with multiple bank accounts	39
no. vendors with multiple bank changes	19

Table 5.6 is a summary of Benford's Law analysis of the first two digits for vendor invoices and payments.

Table 5.6: Control values – Benford's Law

	Invoice Spikes	Payment Spikes
Benford's Law	10	10
	11	11
	19	12
	24	14
	49	18
		19
		22

Experimental values

A detailed trace of this test performed by the prototype is included in Appendix 5 and is discussed below. The dashboard (Figure A5.1) provides a high-level overview of various activities performed in the accounts payable system.

User profiles

A total of **5,996** invoice, **3,185** payment and **792** vendor maintenance activities were recorded during the analysis period (Figure A5.2). Details of each type of activity are provided in Figures A5.3 – A5. 8.

There were **81** active users (*note: sum of users in the table below may add up to more than 81 as some users may have performed more than one type of activity*). They performed the following activities (Table 5.7).

Table 5.7: Activities performed by users

# Transactions	Description of Activity	# Users
5,996	posting document (FB01)	64
3,094	parameters for automatic payment (F110)	2
741	change vendor – accounting (FK02)	5
51	change vendor – centrally (XK02)	10
41	post outgoing payments (FBZ2)	3

Critical combinations

In all, **4 / 81** users were identified as having violated segregation of duties. Details of these violations are provided in Figures A5.9 – A5. 16. Activities performed by these risky users are summarised below (Table 5.8).

Table 5.8: Violation of segregation of duties

# Users	Combination
3	vendor maintenance & invoices
2	vendor maintenance & payments
1	invoices & payments
1	vendor maintenance & invoices & payments

*Note: a user may have made multiple violations

Investigation of user 1USRARSCP

This user was identified as having performed vendor maintenance, invoice entry and payment processing i.e. violating segregation of duties. The user interacted with **40** vendors, performing various activities. These activities were further investigated and are summarised below (Table 5.9).

Table 5.9: Activities performed by user 1USRARSCP

Activity	#	Total value
Bank account changes	19	n/a
Invoices	7	\$ 1,363.13
Payments	38	\$ 5,214,477.05
Round Dollar Invoices	0	0
Round Dollar Payments	4	\$ 710.00
Invoices 5% below approval limit	0	0
Payments 5% below approval limit	0	0
Duplicate transactions	0	0
Vendors touched	40	n/a

In all, 5,996 invoices were entered for a total dollar value of \$10,045,281.90. Anomalous activities include invoices (\$1,363.13) and round dollar invoices (\$710.00). Details of activities performed by this user are provided in Figures A5.17 – 5. 23.

Analysis of vendor activities

There were **3,634** active vendors during the analysis period. The dashboard (Figure A5.1) provides a high-level overview of vendor transactions. Vendor transactions were investigated and the findings are summarised below (Table 5.10).

Table 5.10: Summary of vendor transactions

Activity	Value
number of invoices entered	5,996
number payments processed	3,135
total value of invoices entered	\$ 10,045,281.90
total value of payments processed	\$ 13,226,457.57
top vendor by invoice	0001001516
total value of invoices for this vendor	\$ 1,472,887.36
top vendor by payment	0001001516
total value of payments for this vendor	\$ 1,472,887.36
no. active vendors	3,634
no. vendors sharing bank accounts	131
no. vendors with multiple bank accounts	39
no. vendors with multiple bank changes	19

In all, there were 3,634 active vendors. Anomalous vendor activities include 131 vendors sharing bank accounts, 39 vendors having multiple bank accounts, and 19 vendors having multiple changes to their bank accounts. Details of vendor transactions are provided in Figures A5.24 - A5. 31. Vendor transaction history, a useful tool in detecting flipping of payment details, is shown in Figure A5.36. Visualisation of users interacting with an individual user is shown in Figure A5.37. Visualisation of an individual vendor's transaction history is shown in Figure A5.38.

Findings also indicate that unlike a financial statement audit which is designed to detect material misstatements (Singleton and Singleton 2007), the prototype analyses 'all' activities looking for anomalies associated with fraud symptoms. Investigations performed using the prototype as not susceptible to 'materiality' concerns. The prototype does not consider materiality in its processes or in analysis of audit trail data. This characteristic may improve efficiency and effectiveness of the audit process.

Benford's Law analysis

Benford's Law analysis of the first two digits for vendor invoices revealed spikes at **10, 11, 19, 24** and **49**. Spikes also occurred at **10, 11, 12, 14, 18, 19** and **22** for vendor payments. The largest of these spikes was **49** for invoices (Figures A5.32 and A5.33) and **22** for payments (Figures A5.34 and A5.35).

Conclusion

Verification is an attempt to ensure that the product is built correctly and that the outputs of activities meet specifications imposed on them during the design phase. This test verified that the prototype functioned as planned, and that it met the specifications imposed in the fraud detection framework developed in Chapters 3 and 4. Known anomalies were highlighted and dashboard values corresponded with control values (Table 5.11). It is therefore a logical conclusion that results of subsequent tests performed by the prototype may be dependable. This test also acts as a demonstration of the potential use of this software.

Table 5.11: Correspondence between control values and experimental values

Test	Control value	Experimental value
Activities performed by users	<i>identical</i>	<i>identical</i>
Violation of segregation of duties	<i>identical</i>	<i>identical</i>
Activities performed by user 1USRARSCP	<i>identical</i>	<i>identical</i>
Analysis of vendor transactions	<i>identical</i>	<i>identical</i>
Benford's Law - invoices	<i>identical</i>	<i>identical</i>
Benford's Law - payments	<i>identical</i>	<i>identical</i>

5.2.2. Case study 1a: Data from large international manufacturing company

Analysis period 01/01/2011 to 30/06/2011 – 6 months

(Note: User and vendor names have been masked for confidentiality reasons).

This test exposed the prototype to live data and provides support for research propositions RP2a and RP2b. A detailed trace of this test case study is included in Appendix 6 and is discussed below. The dashboard (Figure A6.1) provides a high-level overview of various activities performed in the accounts payable system.

User profiles

A total of **45,368** invoice, **8,862** payment and **264** vendor maintenance activities were recorded during the analysis period (Figure A6.2). Details of each type of activity are provided in Figures A6.3 – A6. 8.

There were **58** active users. They performed the following activities (Table 5.12).

Table 5.12: Activities performed by users

# Transactions	Description of Activity	# Users
24,690	invoice entry (FB60)	42
20,678	posting document (FB01)	26
8,343	parameters for automatic payment (F110)	24
459	post outgoing payments (FBZ2)	7
255	change vendor – centrally (XK02)	6
68	create vendor – centrally (XK01)	6
60	payment with printout (FBZ4)	5
9	change vendor – accounting (FK02)	2
1	Create vendor – accounting (FK01)	2

Critical combinations

In all, **26 / 58** users were identified as having violated segregation of duties. Details of these violations are provided in Figures A6.9 – A6. 16. Activities performed by these risky users are summarised below (Table 5.13).

Table 5.13: Violation of segregation of duties

# Users	Combination
26	invoices & payments
4	vendor maintenance & invoices
4	vendor maintenance & payments
4	vendor maintenance & invoices & payments

*Note: a user may have made multiple violations

Users **1USRA**, **1USRMI**, **1USREEWAH** and **1USRN** have performed vendor maintenance, invoice entry and payment processing. Activities performed by these users were further investigated and are summarised below.

User **1USRA** performed **11** changes to vendor bank accounts, entered **263** invoices and processed **18** payments (Table 5.14). These activities violate segregation of duties. Performing these activities does not necessarily indicate that fraud has occurred, however, it creates an opportunity for fraud to be perpetrated by this user.

Table 5.14: Summary of activities by 1USRA

Activity	#	Total value
Bank account changes	11	n/a
Invoices	263	\$2,085,287.04
Payments	18	\$298,368.48
Round Dollar Invoices	46	\$471,089.00
Round Dollar Payments	1	\$210,000.00
Invoices 5% below approval limit	2	\$2,861.13
Payments 5% below approval limit	0	0
Duplicate transactions	31	\$9,300,000.00
Vendors touched	32	n/a

In all, 45,368 invoices were entered for a total dollar value of \$186,449,162.56. Anomalous activities include round dollar invoices (\$471,089.00) round dollar payments (\$210,000.00), invoices below approval limit (\$2,861.13) and duplicate transactions (\$9,300,000.00). Potential fraud implications are:

1. round dollar transaction values (invoices and payments) have a higher probability of being fabricated;
2. invoices below an organisations approval limit may indicate attempts at bypassing management review; and
3. duplicate transactions are unexpected and may indicate payment to a fraudster's personal bank account and a subsequent payment to a legitimate vendor (see Table 2.5 and Figure 3.5).

These transactions require further review by internal audit to determine whether they are genuine or fraudulent.

User **1USRMI** performed **37** changes to vendor bank accounts, entered **737** invoices and processed **320** payments (Table 5.15). Anomalous activities include round dollar payments (\$18,136.00), invoices below approval limit (\$2,932.67), payments below approval limit (\$2,915.77) and duplicate transactions (\$8,528.28).

Table 5.15: Summary of activities by 1USRMI

Activity	#	Total value
Bank account changes	37	n/a
Invoices	737	\$5,077,562.14
Payments	320	\$1,285,577.19
Round Dollar Invoices	115	\$3,006,978.00
Round Dollar Payments	5	\$18,136.00
Invoices 5% below approval limit	2	\$2,932.67
Payments 5% below approval limit	2	\$2,915.77
Duplicate transactions	51	\$8,528.28
Vendors touched	147	n/a

User **1USREEWAH** performed **17** changes to vendor bank accounts, entered **2** invoices and processed **3** payments (Table 5.16).

Anomalous activities include invoices (\$53,760.00), payments (\$10,016.84), round dollar invoices (\$53,760.00), and round dollar payments (\$18,136.00).

User **1USRN** performed **263** changes to vendor bank accounts, entered **26** invoices and processed **14** payments (Table 5.17). Anomalous activities include invoices

(\$8,572.18), payments (\$7,304.83), round dollar invoices (\$2,131.00), and round dollar payments (\$30.00).

Table 5.16: Summary of activities by 1USREEWAH

Activity	#	Total value
Bank account changes	17	n/a
Invoices	2	\$53,760.00
Payments	3	\$10,016.84
Round Dollar Invoices	2	\$53,760.00
Round Dollar Payments	2	\$6,010.00
Invoices 5% below approval limit	0	0
Payments 5% below approval limit	0	0
Duplicate transactions	0	0
Vendors touched	20	n/a

Table 5.17: Summary of activities by 1USRN

Activity	#	Total value
Bank account changes	263	n/a
Invoices	26	\$8,572.18
Payments	14	\$7,304.83
Round Dollar Invoices	8	\$2,131.00
Round Dollar Payments	2	\$30.00
Invoices 5% below approval limit	0	0
Payments 5% below approval limit	0	0
Duplicate transactions	0	n/a
Vendors touched	256	n/a

Analysis of vendor activities

There were **1,091** active vendors. The dashboard (Figure A6.1) provides a high-level overview of vendor transactions. Vendor transactions were investigated and the findings are summarised below (Table 5.18).

In all, there were 1,091 active vendors. Anomalous vendor activities include 20 vendors sharing bank accounts, 89 vendors having multiple bank accounts, and 36 vendors having multiple changes to their bank accounts. Details of vendor transactions are provided in Figures A6.17 - A5. 24.

Table 5.18: Summary of vendor transactions

Activity	Value
number of invoices entered	45,368
number payments processed	8,862
total value of invoices entered	\$186,449,162.56
total value of payments processed	\$28,106,039.65
top vendor by invoice	0000030044
total value of invoices for this vendor	\$114,660,580.29
top vendor by payment	0000100027
total value of payments for this vendor	\$2,933,273.73
no. active vendors	1091
no. vendors sharing bank accounts	20
no. vendors with multiple bank accounts	89
no. vendors with multiple bank changes	36

Benford's law analysis

Benford's Law analysis of the first two digits for vendor invoices revealed spikes at **11, 22, 27, 36, 45, 54** and **67**. Spikes also occurred at **22, 27, 36, 37** and **45** for

vendor payments. Other smaller spikes were also observed for invoices and payments. The largest of these spikes was **36** for invoices (Figures A6.25 and A6.26) and **22** for payments (Figures A6.27 and A6.28).

This case exposed the prototype to live data. Data was also collected on processing time and serves as a basis for determining auditor productivity.

5.2.3. Case study 1b: Subset of case study 1a data

Analysis period 01/06/2011 to 07/06/2011 – 7 days

One week of actual transaction data from was processed using the prototype. This test provides data on processing times and additional support for the research propositions. Results of this test also provide support for determining auditor productivity.

A limited trace of this test case study is included in Appendix 7 and is discussed below. The dashboard (Figure A7.1) summarises activities performed in the accounts payable system.

User profiles

A total of **2,021** invoice, **335** payment and **333** vendor maintenance activities were recorded during the analysis period (Figure A7.2).

There were **36** active users. They performed the following activities (Table 5.19).

Table 5.19: Activities performed by users

# Transactions	Description of Activity	# Users
1,235	invoice entry (FB60)	22
786	posting document (FB01)	13
315	parameters for automatic payment (F110)	9
255	change vendor – centrally (XK02)	6
68	create vendor – centrally (XK01)	6
19	post outgoing payments (FBZ2)	2
9	change vendor – accounting (FK02)	2
1	payment with printout (FBZ4)	1
1	Create vendor – accounting (FK01)	2

Critical combinations

In all, **11 / 36** users were identified as having violated segregation of duties. Activities performed by these risky users (Figure A7.3) are summarised below (Table 5.20)

Table 5.20: Violation of segregation of duties

# Users	Combination
8	invoices & payments
3	vendor maintenance & invoices
0	vendor maintenance & payments
0	vendor maintenance & invoices & payments

*Note: a user may have made multiple violations

Analysis of vendor activities

There were **522** active vendors. Vendor transactions (Figure A7.1) are summarised below (Table 5.21).

Table 5.21: Summary of vendor transactions

Activity	Value
number of invoices entered	2,021
number payments processed	335
total value of invoices entered	\$9,104,867.14
total value of payments processed	\$1,433,507.66
top vendor by invoice	0000030044
total value of invoices for this vendor	\$3,200,005.03
top vendor by payment	0000100027
total value of payments for this vendor	\$344,021.55
no. active vendors	522
no. vendors sharing bank accounts	20
no. vendors with multiple bank accounts	89
no. vendors with multiple bank changes	36

Benford's law analysis

Benford's Law analysis of the first two digits for vendor invoices revealed spikes at **13, 18, 22, 27, 36, 42, 45** and **91** (Figure A7.4). Spikes also occurred at **11, 21, 22, 25, 27, 32, 45** and **91** for vendor payments (Figure A7.5). Other smaller spikes were also observed for invoices and payments.

This test was performed on one week's of actual transaction data. The purpose of this test is to provide additional data on processing times in support of determining auditor productivity.

5.2.4. Case study 1a: Summary of findings and recommendations

Given the available data, it appears that support staff **1USRMGR, 1USREAM, 1USRADMIN** and **1USRRTING** are performing functions of normal users –

entering invoices and paying vendors. Several financial transactions having significantly large dollar values have been posted by support staff. This is not recommended practice and violates normal segregation of duties principles (separating users from SAP support functions, and separating entry of invoices/postings and payment functions). This situation presents a considerable fraud risk and requires review by internal audit. It is recommended that posting of financial transactions be restricted to users with relevant authorisations.

Roles of all users that have violated segregation of duties should be reviewed and appropriate restrictions applied to their SAP profiles.

It is generally recommended that users **not use FB01 Post Document** for entry of transactions. This transaction code allows a user to post any financial transaction i.e. general ledger, customer, vendor, inventory, or asset. A user enters a document type (e.g. SA, for GL postings) as part of the header data and then enters relevant data. Security guidelines usually recommend that **no user** be granted access to this transaction code; rather their profile should allow access to the set of specific transaction codes associated with their position, e.g. accounts payable clerk. Access to transaction code **FB01 Post Document** should be restricted. Implementing this restriction will ensure proper segregation of duties.

Several postings with round dollar amounts have been entered. Round dollar values have a higher possibility of being fraudulent. These transactions should be checked to determine whether they are genuine.

Several vendors were found to be sharing bank accounts. These appear to involve vendors with multiple vendor numbers for the same vendor. These vendors should be examined to check that they are genuine. There are also several vendors with multiple bank accounts. These appear to involve vendors with multiple master records. Duplicate vendor master records are a potential fraud risk and should be eliminated. It is recommended that the vendor master file be periodically examined and cleaned to remove any duplicate vendor accounts.

Several cases of 'flipping' of banking details were observed. These should be examined by internal audit to ensure that all bank account changes were authorised.

Benford's Law analysis of the first two digits for vendor invoice and payment transaction revealed several deviations from the expected frequency of these digits. Each of these needs to be investigated to determine the reason for deviation. Large deviations (spikes) are indicative of potential fraud. An investigation was conducted on invoices with spike **36** as this was the largest spike. It was found that several identical amounts have been recorded for the same vendors. These transactions were entered by different users. Similarly, an investigation was conducted on payments with spike **22** as this was the largest spike present in payment data. A comparable pattern of several identical payment amounts recorded for the same vendors was observed (refer to Figures A6.25 and A6.27). These transactions require further investigation by internal audit to determine whether they are genuine, erroneous or fraudulent.

The above findings require close examination by an internal auditor to determine whether these vulnerabilities/anomalies are actually associated with fraudulent activities.

5.3. Processing times

Table 5.22 summarises processing times associated with stage 1 tests (**test data**).

These tests were performed on 1 month of simulated test data using the bench-mark workstation configuration. The majority of the total time is consumed in the data extraction activity. Total time taken for table extraction and pre-processing is 37 minutes and 47 seconds. Time taken to process individual investigations was also tested. This involved selection of individual users or vendors and dynamically producing a series of reports for the selected entity (Figures A4.6, A4.7 and A4.9 to A4.11). Processing times for these activities were all below one minute. Table 5.23 summarises the number of records processed during these tests.

Table 5.22: Processing time - stage 1 test

Activity	Processing Time (h:mm:ss)
Extract SAP tables	0:35:00
Copy text files to system folder	0:01:00
Data conversion	0:00:30
Data import	0:00:10
Set data range	0:00:12
Set approval limits	0:00:10
Pre-process and prepare reports	0:00:40
Extract /Pre-processing Total	0:37:47
Query individual risky user activities	0:00:25
Query individual vendor transactions	0:00:22
Generate Benford's reports	0:00:15
Search vendors for similar names	0:00:13

Table 5.23: Number of records processed

Table Name	# Records
BKPF (accounting document header)	15,281
BSEG (accounting document line items)	27,126
CDHDR (change document header)	2,235
CDPOS (change document line items)	5,182
LFA1 (vendor general data)	58,844

Table 5.24 summarises the processing times associated with stage 2 tests (**case study 1a**). These tests were performed on 6 months of actual transaction data using the bench-mark workstation configuration. Although the tests were performed on a significantly larger time period with more transactions, processing times are almost identical. Table 5.25 summarises the number of records processed during these tests.

Table 5.24: Processing time – stage 2 test

Activity	Processing Time (h:mm:ss)
Extract SAP tables	0:35:00
Copy text files to system folder	0:01:00
Data conversion	0:00:30
Data import	0:00:10
Set data range	0:00:10
Set approval limits	0:00:10
Pre-process and prepare reports	0:00:45
Extract /Pre-processing Total	0:37:45
Query individual risky user activities	0:00:30
Query individual vendor transactions	0:00:22
Generate Benford's reports	0:00:15
Search vendors for similar names	0:00:12

Table 5.25: Number of records processed

Table Name	# Records
BKPF (accounting document header)	112,718
BSEG (accounting document line items)	113,748
CDHDR (change document header)	46,497
CDPOS (change document line items)	301,918
LFA1 (vendor general data)	4,689

Table 5.26 summarises the processing times associated with stage 3 tests (**case study 1b**). These tests were performed on 7 days of actual transaction data using the bench-mark workstation configuration. Again, processing times are almost identical to the previous two tests. Table 5.27 summarises the number of records processed during these tests.

Table 5.28 and Figure 5.4 summarise the number of records processed across all tests. Table 5.29 provides the average processing time for all tests. From this data it can be concluded that processing time is not dependant on the number of records processed by the prototype. Processing time remains comparatively constant regardless of the size of the data-set extracted from the SAP system.

Table 5.26: Processing time – stage 3 test

Activity	Processing Time (h:mm:ss)
Extract SAP tables	0:35:00
Copy text files to system folder	0:01:00
Data conversion	0:00:30
Data import	0:00:10
Set data range	0:00:10
Set approval limits	0:00:10
Pre-process and prepare reports	0:00:40
TIME	0:37:40
Query individual risky user activities	0:00:25
Query individual vendor transactions	0:00:22
Generate Benford's reports	0:00:15
Search vendors for similar names	0:00:12

Table 5.27: Number of records processed

Table Name	# Records
BKPF (accounting document header)	19,896
BSEG (accounting document line items)	20,152
CDHDR (change document header)	414
CDPOS (change document line items)	1,178
LFA1 (vendor general data)	4,689

Table 5.28: Summary of records processed for all tests

Table Name	Test case	Actual case	
	3 months	6 months	7 days
	# Records	# Records	# Records
BKPF	15,281	112,718	19,896
BSEG	27,126	113,748	20,152
CDHDR	2,235	46,497	414
CDPOS	5,182	301,918	1,178
LFA1	58,844	4,689	4,689

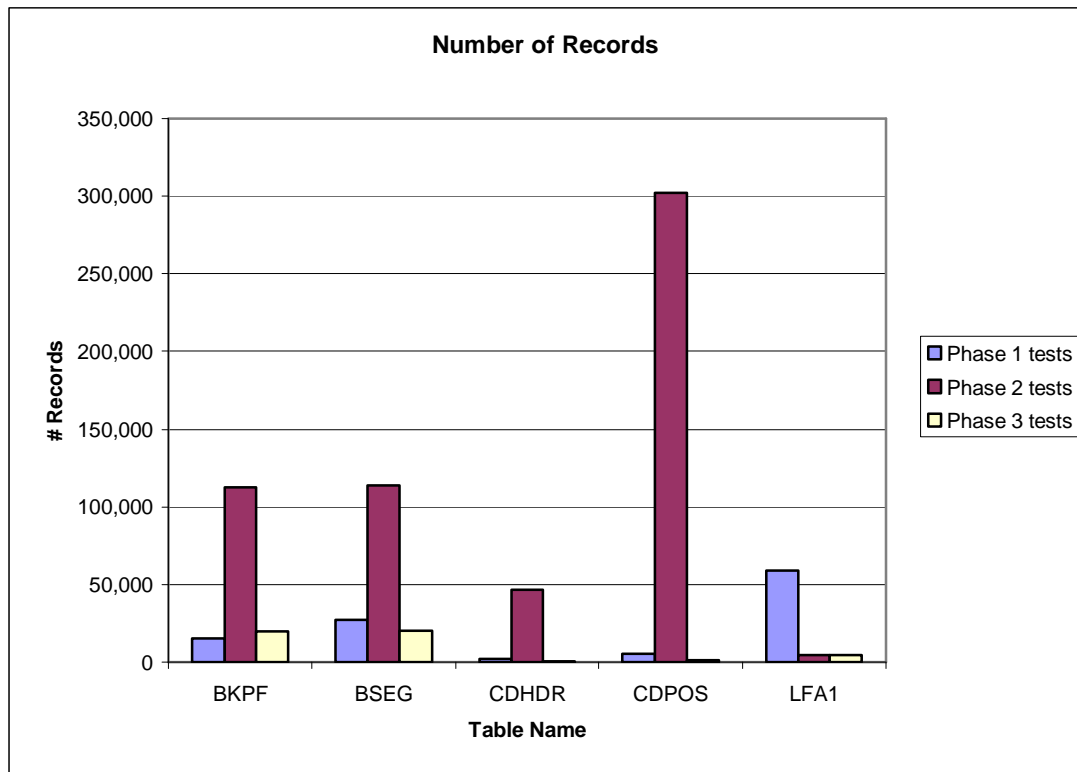


Figure 5.7: Number of records processed

Activity	Test case	Actual case		Average Processing Time (h:mm:ss)
	3 months Processing Time (h:mm:ss)	6 months Processing Time (h:mm:ss)	7 days Processing Time (h:mm:ss)	
Extract SAP tables	0:35:00	0:35:00	0:35:00	0:35:00
Copy text files to system folder	0:01:00	0:01:00	0:01:00	0:01:00
Data conversion	0:00:30	0:00:30	0:00:30	0:00:30
Data import	0:00:10	0:00:10	0:00:10	0:00:10
Set data range	0:00:12	0:00:10	0:00:10	0:00:11
Set approval limits	0:00:10	0:00:10	0:00:10	0:00:10
Pre-process and prepare reports	0:00:40	0:00:45	0:00:40	0:00:42
Extract /Pre-processing Total	0:37:42	0:37:45	0:37:40	0:37:42
Query individual risky user activities	0:00:25	0:00:30	0:00:25	0:00:27
Query individual vendor transactions	0:00:22	0:00:22	0:00:22	0:00:22
Generate Benford's reports	0:00:15	0:00:15	0:00:15	0:00:15
Search vendors for similar names	0:00:13	0:00:12	0:00:12	0:00:12

Table 5.29: Average processing time for all tests

The following conclusions can be justified based on the above test results.

1. Software can be developed to identify potentially fraudulent activities and report these using an intuitive visual interface (RP2a). Stage 1 tests provide evidence that the concept of automated fraud detection is feasible in practice. This is achieved by analysing simulated test data from a SAP enterprise system and producing reports and visualisations identifying potentially fraudulent user behaviours or anomalous activities. An auditor has the additional ability to investigate individual users or vendors in greater detail. In each case, the time taken to query and produce user or vendor centric reports is approximately 30 seconds.
2. Threat monitoring and potential fraud detection can be implemented on a stand-alone external computer system operating independently of an organisation's enterprise system (RP2b). The prototype was developed as a stand-alone application and installed on a separate computer system. Tests were conducted on a variety of data-sets. The prototype was able to handle real data volumes from a real organisation without difficulty. Findings from analysis of case study data revealed that the prototype successfully identified and reported potential threats on a laptop computer, independent of the case study organisation's SAP enterprise system.
3. Efficiency and effectiveness of the audit process can be improved by using technology to perform continuous monitoring of an organisation's enterprise system (RP2c). Experiments were performed using larger and small case study data-sets. Processing time remained comparatively constant regardless of the size of the data-set. Transaction data can be extracted, downloaded, and pre-processed in approximately 40 minutes. An auditor then has the rest of the working day to analyse the data and conduct further detailed investigations of users or vendors. These tests indicate that auditor productivity may be improved when using the prototype to support the audit process. Independent reviews and an expert panel demonstration, discussed in

the following section, provide further evidence in support of this conclusion.

In this section tests were conducted on the prototype to verify whether it satisfied the conditions imposed during the development phase, i.e. that it could identify potentially fraudulent activities in a short time. In the next section, an examination of whether it is the correct product that meets the requirements of the task (i.e. detection of potential fraud) and intended consumer (i.e. an internal auditor) is undertaken.

5.4. Validation and independent review of prototype

Validation is an attempt to ensure that the right product is built, i.e. it fulfils the specific intended purpose and meets the needs of the user. Software evaluation during or at the end of the development process is intended to determine whether it satisfies specified requirements. This is done through dynamic testing and reviews (IEEE 2004 ; Wallace et al. 1996).

Independent reviews were requested from auditing practitioners to obtain feedback on the prototype. In each case, the reviewer(s) were provided with a summary paper (Singh et al. 2011), a one-hour presentation and demonstration of the prototype. The demonstration involved processing and analysing both simulated test data and actual transaction data. The reviews are presented in their entirety in Appendix 9.

The first review was conducted by Mr John Halliday, Executive Director Advisory, BDO, Australia. His comments are stated below:

"A project of this nature is considered to be of high importance to organisations. It provides a mechanism to pro-actively monitor fraud risk, a key risk in any organisation. It also demonstrates a commitment to compliance with Corporate Governance Principles and Recommendations as outlined by ASX Corporate Governance Council.

Automated fraud detection software can provide internal auditors with a tool to efficiently assess the presence of fraud within an organisation. This may also be applied to testing the effectiveness of the controls that management may have in place. A tool of this nature can ensure that the management of the risk of fraud can be undertaken on a more regular or continual basis.

In general, I found the functionality of the tool to be useful. The user interface would require a minimal level of training and some level of understanding of the SAP application, which is a reasonable constraint. The graphs and visualisations clearly communicated a message for the reader. The speed of running the queries was impressive."

The second and third reviews were conducted by K.M., Financial Manager (Internal Audit) and N.J., Financial Director, of the case study site (*note: names have been withheld for confidentiality reasons*). Feedback was requested on the following issues.

1. The importance of such a project for auditing in your organisation.
2. The role that automated fraud detection software could play as an auditing tool for internal auditors.

3. The desirability of a retrospective analysis software tool implemented on a standalone computer system as compared with a system embedded within an enterprise system.
4. The functionality of the prototype, in particular the user interface, reporting and graphical features.
5. Any further comments or suggested improvements to the prototype.

Feedback was also sought from a panel of experts from CPA Australia (Queensland Division - IT Discussion Group) and ISACA⁵ (Queensland Chapter). In total, 20 Certified Practising Accountants (CPAs) constituted the expert panel. A short presentation and demonstration was made to panel members (20 minutes). Members had an opportunity for a hands-on session using the prototype. Their feedback was sought using a survey (Appendix 3) on the following key issues, namely *operation, reporting and visualisations, accuracy and efficiency, and impact on auditor productivity*. They were asked to rate these aspects on a scale of 1 to 7, with 1 being 'Strongly disagree' and 7 being 'Strongly agree'. Twenty three (23) responses were received. A summary of these responses is presented below.

Operation. Panel members rated the prototype as being easy to use, user friendly, and providing adequate on-screen instructions (Table 5.30).

Table 5.30: Operation

Operation <i>(Questionnaire scale 1 to 7)</i>	Mean	Variance	Std Dev.
Easy to use	5.87	0.45	0.81
User-friendly	5.78	0.45	0.67
Navigation of user interface is simple	5.65	0.60	0.78
Onscreen instructions/ help is adequate	5.78	0.36	0.60
Data entry is straightforward	5.70	0.40	0.63
N=23			

⁵ Information Systems Audit and Control Association

Reports and visualisations. Panel members rated reports as being easy to understand, useful in identifying potential fraud and in aggregating enormous amount of information. Visualisations were also seen as enabling identification of relationships or patterns in data that would otherwise be difficult in textual data. Overall, the group rated reports and visualisations as important tools in a fraud investigator's toolkit (Table 5.31 and 5.32).

Table 5.31: Reports

Reports <i>(Questionnaire scale 1 to 7)</i>	Mean	Variance	Std Dev.
Easy to understand	5.91	0.63	0.79
Contains adequate information	5.87	0.48	0.69
Helpful in identifying potential fraud	6.22	0.36	0.60
Are an important tool in a fraud investigators toolkit	6.17	0.33	0.58
N=23			

Table 5.32: Visualisations

Visualisations (charts & diagrams) <i>(Questionnaire scale 1 to 7)</i>	Mean	Variance	Std Dev.
Easy to understand	5.87	0.87	0.92
Useful in aggregating an enormous amount of information	6.09	0.54	0.73
Enables effective exploration of data in a graphical format	6.13	0.57	0.76
Enables identification of relationships or patterns in data that are otherwise difficult to do in textual data	6.17	0.60	0.78
Enhances investigation and analysis for potential fraud	6.22	0.54	0.74
Are an innovative way of presenting information	6.35	0.42	0.65
Are an important tool in a fraud investigators toolkit	6.04	0.77	0.88
N=23			

Accuracy, efficiency and performance. The prototype was rated as producing quality, useful and accurate results. Panel members agreed that the prototype was an improvement over basic analytical tools and that results were produced in a much faster time than if done manually. They also felt that there was potential to save costs and reduce future fraud by early detection of suspicious user activities (Table 5.33).

Table 5.33: Accuracy, efficiency and performance

Accuracy, Efficiency and Performance <i>(Questionnaire scale 1 to 7)</i>	Mean	Variance	Std Dev.
Produces quality results that are useful in identifying potential fraud	5.96	0.50	0.71
Results are accurate and dependable	6.09	0.36	0.60
Produces the same results as a human expert	6.00	0.55	0.74
Generates results much faster than doing a similar task manually	6.35	0.42	0.65
Is an improvement over basic analysis as it replaces blind querying of data with contextual analysis	5.96	0.59	0.77
Significantly enhances the internal auditing process	5.87	0.30	0.55
Potential to save costs due to improved fraud detection	6.13	0.39	0.63
Potential to reduce future fraud by early detection of suspect user activity	6.22	0.45	0.67
N=23			

Auditor productivity. Panel members collectively agreed that the prototype may reduce time taken to identify potential fraud (Table 5.34). They were asked to rate the number of person days it would take to review a system for fraud based on 30,000 transactions. Their responses are shown in Tables 5.35 to 5.37.

Table 5.34: Auditor productivity

Auditor Productivity <i>(Questionnaire scale 1 to 7)</i>	Mean	Variance	Std Dev.
This software may reduce time taken to identify potential fraud in an organisation	6.30	0.49	0.70
N=23			

Panel members generally agreed that it would take 20+ days (39.1%) or it would be an impractical task (60.9%) to audit the stated number of transactions, if done manually (Table 5.35).

Table 5.35: Time to process data manually

Q 4.1a	Value (days)	Count	%
How long would it take to review for fraud, if done manually?			
Impractical	U	14	60.9%
	20+	9	39.1%
	10	0	0.0%
	5	0	0.0%
	3	0	0.0%
	1	0	0.0%
Less than 1	<1	0	0.0%
N=23			

Table 5.36: Time to process data with other software

Q 4.1b	Value (days)	Count	%
How long would it take to review for fraud, if done using other software?			
Impractical	U	0	0.0%
	20+	8	34.8%
	10	9	39.1%
	5	3	13.0%
	3	1	4.3%
	1	2	8.7%
Less than 1	<1	0	0.0%
N=23			

Panel members agreed that it would take between 1 and 20+ days to audit the stated number of transactions using other software (i.e. ACL, Access, Excel, etc.) (Table 5.36).

Panel members agreed that it would take between <1 to 5 days to audit the stated number of transactions using the prototype. Most agreed that 3 days (43.5%) was standard, 21.7% said 1 day and 17.4% said either 5 days or <1 day (Table 5.37). From these ratings, it may be concluded that using the prototype as a tool for detection of potential fraud improves auditor productivity.

Table 5.37: Time to process data with prototype

Q 4.1c	Value (days)	Count	%
How long would it take to review for fraud, if done using the prototype software?			
Impractical	U	0	0.0%
	20+	0	0.0%
	10	0	0.0%
	5	4	17.4%
	3	10	43.5%
	1	5	21.7%
Less than 1	<1	4	17.4%
N=23			

Table 5.38: Overall evaluation

Overall Evaluation <i>(Questionnaire scale 1 to 7)</i>	Mean	Variance	Std Dev.
This software represents substantial advances over other tools currently available in the market	5.96	0.41	0.64
If available, I am likely to use this software	5.70	0.68	0.82
If available, I am likely to recommend this software to others	6.04	0.59	0.77
Overall, this software is a useful auditing tool	6.22	0.72	0.85
N=23			

Overall evaluation. Panel members considered the prototype a useful auditing tool that represented substantial advances over other tools currently available in the market. They are likely to use or recommend this tool should it be commercially available (Table 5.38).

Panel members were asked to provide their comments on the following matters.

i) Features of the software that were useful

Selected comments are given below.

"Visualisations and drill-downs are good."

"Segregation of duties looks good. Flipping looks good."

"Dashboard a great idea."

"Ability to find out details from the dashboard within a few seconds."

"Ability to do different views."

"Identifying user behaviour and eliminating patterns that don't breach SoD."

"Useful in fraud prevention and identification in [accounts payable]."

"A great framework to begin to look at [accounts payable]."

ii) Features of the software that could be improved

Selected comments are given below.

"Broaden scope to include other areas such as [accounts receivable], [general ledger], [sales and distribution], [inventory], etc."

"Detect collusion."

"Identify missing fields in data entered by users."

Most panel members agreed that the dashboard and visualisations were especially useful in promptly identifying activities in an accounts payable system. They also agreed that it was an excellent tool for fraud prevention and detection. Some commented that the prototype should be extended to include other areas of the accounting cycle including accounts receivable, general ledger and inventory systems (these comments are addressed in section 5.6).

5.5. Prototype implementation and testing and propositions

This Chapter addresses the following three research propositions.

RP2a: Software can be developed to identify potentially fraudulent activities and report these using an intuitive visual interface.

Prototype software is developed on a Windows-based computer system. The user interface is web browser based. Aspects of the web programs are written in VBScript. Visualisation components are written in GraphViz v1.01. The web based user interface provides a simple, intuitive, graphical interface that most users are already familiar with. Accounting audit trails are routinely extracted from the SAP system and imported into the prototype for pre-processing and analysis. Extracted data tables are imported, cleansed (to remove any inconsistencies) and preformatted before being pre-processed by the prototype. On completion of pre-processing, reports and visualisations are produced. Thus, it is established that software can be developed to identify potentially fraudulent activities and report these using an intuitive visual interface. RP2a is therefore supported.

RP2b: Threat monitoring and potential fraud detection can be implemented on a stand-alone external computer system operating independently of an organisation's enterprise system.

Prototype software is developed as a stand-alone application and installed on a separate computer system. Tests are conducted on a variety of data-sets. The prototype is able to handle real data volumes from a real organisation without difficulty. Findings from analysis of case study data reveals that the prototype successfully identifies and reports potential threats on a laptop computer, independent of an organisation's SAP enterprise system. Feedback received from independent reviews and expert panel members indicate support for a stand-alone prototype. Thus, it is established that threat monitoring and potential fraud detection can be implemented on a stand-alone external computer system. RP2b is therefore supported.

RP2c: Efficiency and effectiveness of the audit process can be improved by using technology to perform continuous monitoring of an organisation's enterprise system.

Experiments are performed with the prototype to provide information on processing times. Results indicate that processing times remain comparatively constant regardless of the size of the data-set. Transaction data can be extracted, downloaded, and pre-processed in approximately 40 minutes. An auditor then has the rest of the working day to analyse the data and conduct further detailed investigations of users or vendors. These tests indicate that auditor productivity may improve when using the prototype to support the audit process. Feedback from the panel of experts indicate that the prototype is capable of producing results in a very short time period and that there is potential to save costs and reduce propensity for future fraud due to early detection. Thus, it is established that efficiency and effectiveness of the audit process may be improved by using technology to perform continuous monitoring. RP2c is therefore supported.

5.6. Conclusion

This Chapter addresses research propositions RP2a, RP2b and RP2c. Implementation and test results of the prototype are described with reference to a number of Appendices. Results indicate that the following:

- i) Software can be developed to identify potentially fraudulent activities and report these using an intuitive visual interface. The reporting system provides an auditor with a simple, intuitive web-based interface that permits effective interrogation of user and vendor activities.

- ii) Threat monitoring and fraud detection can be effectively implemented on a stand-alone external computer system operating independently of an organisation's enterprise system. The prototype was successfully developed as a stand-alone application and installed on a separate computer system.
- iii) Efficiency and effectiveness of the audit process can be improved by using technology to perform continuous monitoring of an organisation's enterprise system. Experimentation with case study data and feedback from experts indicate that auditor productivity may be improved when using the prototype.

Verification of the prototype was achieved by performing a series of tests using test data involving simulated activity over a period of one month. These tests assessed the functionality of the prototype to ensure that it met the specifications imposed in the fraud detection framework developed in Chapters 3 and 4. Known anomalies were highlighted and dashboard values corresponded with control values. It was concluded that results of subsequent tests performed by the prototype may be dependable.

Case study data from a large international manufacturing was processed using the prototype. These tests exposed the prototype to live data and served to provide support for research propositions RP2a and RP2b.

Validation was achieved by requesting independent reviews from auditing practitioners and an expert panel demonstration. Feedback was very positive and indicated support for the prototype. Panel members felt that the dashboard and visualisations were especially useful in promptly identifying anomalies. They also believed that performing analysis on large transaction data-sets (i.e. 30,000 or more)

is not practical if done manually. They agreed that the prototype was capable of producing results in a very short time period and that there was a potential to save costs and reduce propensity for future fraud due to early detection. An expert independent reviewer stated, "...*the speed of running queries was impressive*". Panel members also indicated that there would be support for such a product if it was commercially available.

Timing experiments were conducted using the prototype. They indicated that processing time remained comparatively constant regardless of the size of transaction data. The prototype was able to handle real data volumes from a real organisation without difficulty. Comments received from the case study organisation indicated that they were impressed with the prototypes ability to highlight anomalies in the dashboard within a few seconds of processing. Processing time estimates indicate that transaction data can be extracted, downloaded, and pre-processed in approximately forty minutes. An auditor then has the rest of the working day to analyse the data and conduct further detailed investigations of users or vendors. These tests provide evidence that auditor productivity may improve when using the prototype as a tool for detecting the possibility that fraud has occurred.

CHAPTER 6

Conclusion and Further Research

6.0. Introduction

This research seeks to answer the question whether *a generalised model for proactive fraud detection in enterprise systems can be developed*. The approach adopted was to develop prototype software to provide evidence that the concept of proactive fraud detection in enterprise systems is feasible in practice. Conclusions from results of this research are summarised in this Chapter. Contributions from these results to the study of proactive fraud detection in enterprise systems are described. Perceived limitations of the prototype are discussed and potential extensions to the current prototype are identified. Opportunities for further research are also discussed.

6.1. Summary of results from this study

The primary objective of this research is to explore and develop innovative methods for proactive detection of potential fraud in enterprise systems. The intention was to build a model for fraud detection based on analysis of patterns or signatures. This objective is addressed by answering the main question. Two research sub-questions and five research propositions were formulated. The results achieved pertaining to these research propositions are summarised below.

SQ1: How do enterprise systems support proactive detection of potential fraud in financial transactions?

This question examined the possibility of detecting fraudulent activities in an enterprise system. To address this sub-question, three propositions were formulated.

RP1a: Enterprise system audit trails document adequate data to allow retrospective monitoring of user activities.

SAP audit trails were examined. It was established that they provide detailed descriptions of functions performed by users within an enterprise system. Each function has a transaction code associated with it (for example, FB60 – Enter Vendor Invoice). Each transaction code executed by a user is recorded in the audit trail (Best 2000). Audit trail data is stored in several tables within the SAP enterprise system (Figure 4.3). This data documents **changes to master records** and **accounting audit trails**.

Changes to master records are stored in two tables (Figures 4.4 and 4.5) namely **CDHDR** (Change Document Headers) and **CDPOS** (Change Document Items) (Padhi 2010 ; Best et al. 2009 ; Hirao 2009 ; Best 2005). Changes to master records include creation and deletion of master records and changes to fields (for example, FK01 - Create Vendor Master Record, FK02 - Change Vendor Master Record). Thus it is possible to identify an individual user making these changes.

Accounting audit trails are stored in tables **BKPF** (Accounting Document Header), **BSEG** (Accounting Document Line Item), **SKAT** (General Ledger Account Texts), and **LFA1** (Vendor General Data) (Figures 4.6 to 4.9). Tables BKPF and BSEG store posting histories for both general ledger and subsidiary ledgers. This facilitates integration of data and automatic reconciliation of subsidiary ledgers with control accounts. General ledger account texts (names) are stored in table SKAT. Vendor general data including vendor name, date created and creating user are stored in table LFA1 (Best et al. 2009). Thus it is possible to identify an individual user performing these activities.

It was established that enterprise system audit trails document adequate data to allow retrospective monitoring of user activities. Thus RP1a is supported.

RP1b: Violations in segregation of duties can be identified by analysing audit trails for critical combinations of user activities.

This study supports the following principles of segregation of duties within the accounts payable (AP) function as proposed by Little and Best (2003) (Figure 3.11).

SoDs Principle 1: Users who can create and modify vendor master records should not be able to post accounting transactions.

SoDs Principle 2: Payments should be performed by someone other than the person who enters vendor invoices.

Evidence supporting these principles was obtained by examining data from SAP tables **CDHDR** (Change Document Headers) and **CDPOS** (Change Document Items), **BKPF** (Accounting Document Header), **BSEG** (Accounting Document Line Item), and **LFA1** (Vendor General Data). It was determined that this data allows association of actions with users'. Critical combinations of user activities have been designed in section 4.4.1. Table 4.3 and 4.4 lists the combination of activities a user has to perform in order to violate each of the SoDs principles. If any of these violations are identified then further investigation of the offending user's activities is necessary to determine whether any fraudulent transactions have been performed. Therefore it is feasible to detect violations in segregation of duties with available data.

It was established that violations in segregation of duties can be identified by analysing audit trails for critical combinations of user activities. Thus RP1b is supported.

RP1c: Potentially fraudulent transactions can be identified by investigating user activities that violate segregation of duties, match known fraud symptoms, or appear otherwise anomalous.

Given the ability to identify violations in segregation of duties, it is feasible to detect potentially fraudulent transactions made possible by these violations. For example, the ability to identify users who have changed vendor details, entered an invoice and paid the invoice permits detection of potential accounts payable fraud. In addition, further potential fraud can be detected through examination of other anomalous

activities. A catalogue of known fraud symptoms informs this process (Table 4.5).

The following analyses are available to an auditor.

- i). *Bank account 'flipping'* – checks for changes to banking details, a change back to original, with transactions processed in the interim period.
- ii). *Duplicates test* - checks for any duplicates, for example, invoices, payments or vendors.
- iii). *Trend analysis* - compares activities over two or more periods, to identify variances over time, for example, vendors with minimal payments in prior periods but large payments in current period may be fraudulent payments.
- iv). *Benford's Law* - gives expected frequencies of digits in numerical data. Spikes may be indicative of fraud and require further investigation.
- v). *Stratification* - identifies the number and dollar value of vendor payments that occur within a specified interval, for example 5% below an approval limit.
- vi). *Graphing* - provides a visual means of documenting anomalous activities.

It was established that potentially fraudulent transactions can be identified by investigating user activities that violate segregation of duties, match known fraud symptoms, or appear otherwise anomalous. Thus RP1c is supported.

SQ2: How can detection of potential fraud in enterprise systems be effectively and efficiently automated to facilitate auditor productivity?

This question examined the possibility of automating fraud detection in an enterprise system. To address this research sub-question, three propositions were formulated.

RP2a: Software can be developed to identify potentially fraudulent activities and report these using an intuitive visual interface.

Prototype software was developed using an Intel Core i5-based computer system running the Windows 7 operating system. The primary development environment was SAS v9.2 for Windows. The user interface was web browser based and required Microsoft Internet Explorer 5.0 or above. Aspects of the web programs were written in VBScript. Visualisation components were written in GraphViz v1.01. The intention of the prototype was to demonstrate the feasibility of the concept of proactive fraud detection in practice. The web based user interface provided a simple, intuitive, graphical interface that most users were already familiar with.

Data requirements for fraud detection in SAP enterprise systems have been discussed in Section 4.3.1. Accounting audit trails were routinely extracted from the SAP system and imported into the prototype for pre-processing and analysis. SAP data tables were accessed through the SAP menu or by using transaction codes (Table 5.1). Extracted data tables were imported, cleansed (to remove any inconsistencies) and preformatted before being pre-processed by the prototype (Figure A4.16). On completion of pre-processing, reports and visualisations were produced. Further detailed investigation of user (Figure A4.6) or vendor (Figure A4.9) was also possible.

The following experiments were conducted using the prototype.

1. Test data - the system was tested using a set of test data involving simulated activity over a period of one month. This test served to assess whether the software performed correctly, and that it met the specifications imposed in the fraud detection framework developed in Chapters 3 and 4. Based on tests run, the prototype did meet expectations.
2. Case study - six months of actual transaction data was processed using the prototype. This data was obtained from a large international manufacturing company. These tests exposed the prototype to live data. Again, the prototype met expectations and identified potential frauds for future investigations.

It was established that software can be developed to identify potentially fraudulent activities and report these using an intuitive visual interface (See Figure A5.1). Thus RP2a is supported.

RP2b: Threat monitoring and potential fraud detection can be implemented on a stand-alone external computer system operating independently of an organisation's enterprise system.

Prototype software was developed as a stand-alone application and installed on a separate computer system. Tests were conducted on a variety of data-sets. The prototype was able to handle real data volumes from a real organisation without difficulty. Findings from analysis of case study data revealed that the prototype successfully identified and reported potential threats on a laptop computer,

independent of the case study organisation's SAP enterprise system. The following comments were received from the case study organisation:

"It is an advantage that we can operate this software on a standalone computer system rather than embedded in our main SAP system as it minimizes the disruptions to routine operations and allows retrieving reports at any given time even when the on-line system is not available."

It was established that threat monitoring and potential fraud detection can be implemented on a stand-alone external computer system. Thus RP2b is supported.

RP2c: Efficiency and effectiveness of the audit process can be improved by using technology to perform continuous monitoring of an organisation's enterprise system.

The following experiments were conducted using the prototype to provide information on processing times

1. Case study 1a - six months of actual transaction data - to determine the time it would take to process a large data-set.
2. Case study 1b - one week of actual transaction data – to provide a comparative analysis of the time it would take to process a small data-set.

Processing time remained comparatively constant regardless of the size of the data-set. Transaction data can be extracted, downloaded, and pre-processed in approximately 40 minutes (Figure 5.3). An auditor then has the rest of the working day to analyse the data and conduct further detailed investigations of users or vendors. These tests indicated that auditor productivity could be improved when

using the prototype to support the audit process. The following feedback was received from the case study organisation.

"As one of Asia's⁶ largest companies who is operating in a SAP environment, it is extremely vital to have system based controls on detecting and preventing fraudulent activities. Considering the number of transactions that take place every day, it has become impractical to check each transaction in detail manually unless they are covered by way of controls in place. In such an environment this software will immensely help our internal auditors to carry out various tests in detecting frauds and errors."

Feedback from the panel of experts indicated that the prototype is capable of producing results in a very short time period and that there is potential to save costs and reduce propensity for future fraud due to early detection. They were also impressed with the prototypes ability to highlight anomalies in the dashboard within a few seconds of processing. Findings also indicate that unlike a financial statement audit which is designed to detect material misstatements (Singleton and Singleton 2007), the prototype analyses 'all' activities when identifying anomalies associated with fraud symptoms. Investigations performed using the prototype are not susceptible to 'materiality' concerns. The prototype does not consider materiality in its processes or in analysis of audit trail data. This characteristic may improve efficiency and effectiveness of the audit process.

It was established that efficiency and effectiveness of the audit process can be improved by using technology to perform continuous monitoring. Thus RP2c is supported.

⁶ Name of location has been changed to maintain confidentiality.

These results support the conclusion that software can be developed to identify potentially fraudulent activities and report these using an intuitive visual interface that permits effective interrogation of user and vendor activities. Threat monitoring and fraud detection can be effectively implemented on a separate standalone computer system. Auditor productivity could be improved when using the prototype to support the audit process. Feedback obtained from the panel of experts was very positive and indicated support for the prototype. They collectively agreed that using the prototype may reduce time taken to identify potential fraud. They also felt that there would be support for such a product if it was commercially available. Mr. John Halliday, Executive Director Advisory, BDO, Australia commented that;

"Automated fraud detection software can provide internal auditors with a tool to efficiently assess the presence of fraud within an organization. This may also be applied to testing the effectiveness of the controls that management may have in place. A tool of this nature can ensure that the management of the risk of fraud can be undertaken on a more regular or continual basis."

Thus, it can be concluded that *a generalised model for proactive fraud detection in enterprise systems can be developed*, thereby providing support for the primary research question.

6.2. Contributions

There are a number of contributions this study makes to both the literature and auditing practice. The following sub-sections highlight the contributions of this research to the development of fraud detection models and how it can be applied to real-world organisations.

6.2.1. Theoretical contributions

The research problem addressed in this study is not new. The first recorded financial fraud was perpetrated by a Greek merchant in 300 B.C. (Beattie 2009). This study, therefore, intended to determine the feasibility of using technology to proactively detect potential fraud in enterprise systems. As business processes become more complex and integrated, and organisations implement large-scale enterprise systems, the need for proactive fraud detection becomes a necessity. Furthermore, fraudsters are becoming more sophisticated in their use of technology and in their ability to commit and conceal fraudulent activities. Several gaps in the literature relating to fraud detection are identified in Chapter 2. The contributions made by this study address these gaps and add to the understanding of fraud detection in enterprise systems, in general, and SAP, in particular.

There appears to be limited research in continuous monitoring and fraud detection (Du and Roohani 2007 ; Kuhn Jr and Sutton 2010). Furthermore, there appears to be no published research in developing a generalised model for proactive detection of potential fraud in enterprise systems. This study develops and tests a conceptual model for proactive detection of potential fraud in enterprise systems. The model recommends the following process: i) identify the types of frauds that can occur; ii) catalogue the fraud symptoms; iii) use computer technology to detect fraud symptoms; and iv) analyse the results (Figure 3.3). A Monitoring and Control Layer (MCL) based model was recommended as it has the following inherent characteristics:

- i). It is not limited to sampling a subset of an organisation's transactions, as is the case with the traditional manual audit. Therefore there is no sampling risk.
- ii). It provides frequent opportunities for identifying potential fraud. This will most likely lead to a reduction in the time taken to detect fraud, from several months to days or hours.
- iii). In-built data analytics will assist in determining the propensity for frauds occurring in the future. This attribute may be used in identifying, and proactively correcting deficiencies in internal controls thereby leading to a possible reduction in fraud in an organisation.
- iv). Access to an enterprise system is only required for data extraction purposes. A separate system is used for continuous monitoring and analysis. Impact on the enterprise system being monitored is therefore negligible as there is no overhead of running additional software. Independence of auditors' computer systems is also preserved.

The conceptual model developed in this study characterises i) the fundamental nature of fraud, and ii) its detection. Firstly, it identifies mental states that fraudsters experience prior to perpetrating a fraud. Once a fraudster determines *what to steal*, the next decision is *how to steal it* (Figure 2.8). A fraudster subsequently decides on a specific fraud method to achieve the desired outcome of; i) perpetrating a fraud and, ii) concealing it to avoid detection (Figure 2.9). Secondly, the model focuses on detection of potential fraud in an organisation. This is achieved by:

- i). creation of a catalogue of fraud symptoms;
- ii). translation of fraud symptoms into detection strategies;
- iii). design and development of a prototype;
- iv). experimentation with enterprise system data; and
- v). validation of prototype by an expert panel.

In summary, this study contributes to the literature by developing a conceptual model for proactive detection of potential fraud in enterprise systems.

Prior research on continuous auditing does not appear to deliver a model that allows an audit to be carried out proactively and continuously without difficulties (Hunton et al. 2004). Additional study is required to develop approaches of continuous auditing that are specifically applicable to auditing of financial transactions in enterprise systems. When considering an automated solution for proactive fraud detection, the focus has to be on questions that can be answered with the aid of computerised tools (Lanza 2007). Some questions are too subjective, for example, "Are the vendor's goods or services of good quality?" Any effort to develop an automated solution requires evidence that i) is documented in an enterprise systems audit trails, and ii) can be investigated using data analytics tools. The model developed in this study makes a contribution to proactive detection of potential billing fraud schemes involving shell companies and non-accomplice vendors. Detection methods are grouped into two categories: i) critical combinations of user activities; and ii) known fraud symptoms.

Many frauds occur because fraudsters exploit the lack of internal controls or they may override existing internal controls that are poorly implemented. Users with the

ability to violate segregation of duties are in a position to perpetrate fraud. Identifying these users is an important first step. Subsequent further analysis of the activities performed by these offending users may be helpful in identifying potentially fraudulent activities. For example, a user that creates or modifies a vendor master record should not be able to enter an invoice. Having this capability does not indicate that a fraud has taken place, but it does create an opportunity for a fraud to be perpetrated. By detecting these critical combinations of user activities i) an auditor can further investigate transactions that match known fraud symptoms, or appear otherwise anomalous, and ii) an organisation can take steps to correct the situation thereby reducing the possibility of future fraud. This study contributes a catalogue of methods to detect symptoms of known frauds schemes (Table 3.2) and data sources required to detect these fraud symptoms in the SAP enterprise system (Table 4.1 and Figures 4.3 to 4.9). SAP audit trails provide detailed descriptions of functions performed by each user. Every SAP function has a unique transaction code (t-code) associated with it. Critical combinations may be identified by examining t-codes of functions performed by users. This study contributes a list of t-codes pertinent to each of the SoDs principles previously discussed (Tables 4.2 to 4.4). Users that perform these combinations are identified as having violated segregation of duties principles. Their activities require further investigation to determine whether they match known fraud symptoms, or appear otherwise anomalous. This study contributes a catalogue of known fraud symptoms that informs this process (Table 4.5). Further contributions include design specifications and algorithms for proactive detection of potential fraud as discussed in section 4.5.

Enterprise system audit logs are not enabled by default. Furthermore, they may be turned off at any time to conserve storage space (SAP-AG 2009). Any system designed to detect potential fraud may be rendered ineffective should audit logs be disabled or turned off. An important contribution is that the model developed for this study uses actual enterprise system transaction data for detection of potential fraud, instead on relying on system generated audit logs. In doing so it avoids situations where audit logs may be turned off.

In summary, this study makes a contribution by developing a model for proactive detection of potential billing fraud schemes involving shell companies and non-accomplice vendors. It contributes a catalogue of methods to detect symptoms of known frauds schemes and data sources required to detect these fraud symptoms in SAP. It also contributes a list of transaction codes required to detect violation of each of the principles of segregation of duties and known fraud symptoms in SAP. Further contributions include design specifications and algorithms for proactive detection of potential fraud in SAP.

Research is required in the development of innovative approaches to continuous fraud detection in organisations that use enterprise systems, and to demonstrate how this can be done efficiently and effectively (Rezaee et al. 2002). Information overload from alerts when implementing continuous fraud detection systems (Alles et al. 2008 ; Alles et al. 2006 ; Kuhn and Sutton 2006) appears to be a problem. Integrity of the data used for continuous fraud detection is of concern (Kuhn Jr and Sutton 2010). Enterprise systems generate hundreds of thousands to millions of transactions annually. While most of these are legal and routine transactions, a small number may be fraudulent. The enormous amount of generated transactions makes it

difficult to find these few instances among legitimate transactions. For a large organisation, this means monitoring hundreds of thousands of transactions and then investigating suspicious ones in depth at considerable expense. The problem becomes overwhelming and is growing worse (Chang et al. 2007). One approach is to use visualisation to present information graphically (Fetaji 2011 ; Liang and Miranda 2001).

Visualisation is a general term used to describe any technology that enable users to 'see' information in order to help them better understand and put it in an appropriate context (TechTarget 2010 ; GraphViz 2010). Visualisation tools go beyond the standard charts and graphs, displaying data in more sophisticated ways such as dials and gauges, heat maps, tree maps and detailed bar and pie charts. Patterns, trends and correlations that might go undetected in text-based data can be exposed and recognised easier with visualisation.

This study makes a contribution to literature by developing and implementing methods for visualising user activities in an enterprise system's transaction data. These visualisation methods serve to reduce the problem of *information overload* by presenting voluminous information graphically. For large organisations that generate an enormous amount of information daily, of which only a small percentage may be fraudulent, these visualisation methods may assist in highlighting suspicious activities with minimal effort.

6.2.2. Contributions to the practice of fraud detection

In practice, this study makes contributions by developing prototype software for detection of potential fraud. The prototype is an implementation of the concepts discussed in the theoretical contributions above. The intention is to improve auditor productivity developing software to support the auditing function in an effective and efficient way.

Visualisation is an important aspect in reducing *information overload*. The prototype produces a combination of user- and vendor-centric reports and visualisations. A *Fraud Analytics Dashboard* provides a high-level overview of activities performed in the system (Figure A5.1). Transaction activities are summarised using pie and bar charts (Figure A5.32) and link node diagrams (Figure A5.12). These presentation methods augment standard reports produced by the prototype and support a reduction in information presented to an auditor.

Another contribution is the practical implementation of Benford's Law. This law gives expected frequencies of digits in numerical data. Spikes may be indicative of potential fraud and require further investigation. The prototype analyses invoice and payment data and produces visualisations in the form of vertical bar charts. By plotting actual transaction data frequencies against Benford's expected frequencies, an auditor can visually inspect and identify anomalies promptly (Figure A5.32).

The prototype's user-interface is web browser based. All user-interaction, including access to reports and visualisations, occur via the web browser. Usability is a main characteristic of the web browser interface as it ensures that the prototype is used for its intended purpose by its target audience efficiently and effectively. It provides a

simple, intuitive and user friendly interface that is easy to learn and use as most users are already familiar with.

The prototype is developed as a stand-alone application and installed on a computer system independent of an organisation's enterprise system. Being independent ensures that an auditor can perform analytics at any time without requiring access to an enterprise system. Integrity of data used for the auditing process, as well as the process itself, is maintained as these tests are performed independently of an organisation and its employees.

Feedback from the panel of experts indicate that the prototype is an improvement over basic analytical tools (such as Microsoft Excel, Microsoft Access, ACL, etc.) and results are produced in a much faster time than if done manually. They consider the prototype as being easy to use and user friendly. The dashboard and visualisations are especially useful in promptly identifying potentially fraudulent activities. Participants commented that visualisations enable identification of relationships or patterns in data that would otherwise be difficult in textual data. They stated that the prototype may potentially save costs and reduce future fraud by early detection of suspicious user activities.

In summary, this study contributes prototype software to the practice of fraud detection. The prototype implements concepts such as the catalogue of fraud symptoms, visualisations and Benford's Law analysis to detect potential fraud in enterprise systems. Feedback from expert panel members and the case study organisation indicate that the prototype effectively and efficiently identifies

potentially fraudulent activities and that it improves auditor productivity. The following comment was received from Mr John Halliday, Executive Director Advisory - BDO, Australia.

In general I found the general functionality of the tool to be useful. The user interface would require a minimal level of training and some level of understanding of the SAP application, which is a reasonable constraint. The graphs and visualisations clearly communicated a message for the reader. The speed of running the queries was impressive."

As an exploratory study, the findings and contributions need to be considered within the limitations of the study.

6.3. Limitations

The first limitation of this study is that there is insufficient access to data to determine the level of fraud prevalent in organisations. Many frauds that occur are handled quietly by the victim organisations as they are more concerned about the embarrassment of making frauds public and the costs associated with fraud investigations. Consequently, organisations with and without fraud experiences are not prepared to provide access to their transaction data. This situation is confirmed in a survey conducted by AuditNet (2011) where it was found that one of the 10 key challenges for data analytics is the difficulty of getting data to perform analyses. Therefore, the single case study approach was adopted for this study. Data from the same case was investigated to determine how it changed over time and whether these changes are indicative of potentially fraudulent activities. A concern often expressed with single case designs is the ability to generalise from a single case. Multiple case

designs are preferable as the evidence from multiple cases are considered more compelling (Yin 2009). Several unsuccessful attempts were made to access more case studies. Limited access to organisations transaction data made it impractical to follow an inductive approach to the design of proactive fraud detection tools.

The second limitation of this study is the generalisability of results is limited. The focus of this study is on a single category of occupational fraud, namely, asset misappropriation. Within asset misappropriation, the study focuses on billing fraud schemes involving shell companies and non-accomplice vendors in accounts payable. This limits identification of potential threats or frauds. Generalising the findings to other categories of fraud (such as accounts receivable) therefore must be made with caution.

A third limitation of this study is that the prototype was validated by a limited number of experts. Twenty Certified Public Accountants, all members of CPA Australia (Queensland Division - IT Discussion Group) and ISACA⁷ (Queensland Chapter) participated in the expert panel. Three independent reviews were also conducted by auditing practitioners. The voluntary nature of participation may refer to a situation where only those interested in *proactive fraud detection* chose to attend. Comments and views expressed regarding validation of the prototype are based on feedback obtained from these experts. Generalisability of this feedback must therefore be made with caution.

A fourth limitation of this study is that the prototype may incorrectly identify anomalous activities. Type I and Type II errors were discussed in Section 4.8.

⁷ Information Systems Audit and Control Association

Failing to correctly profile users, identify users that violate segregation of duties, and identify anomalous vendor transactions, including unusual activities related to vendor payments, is of primary concern. Complete segregation of duties may not be feasible in organisations with small accounting sections. In these instances, the prototype may incorrectly identify legitimate activities as anomalous or vice versa. It is therefore necessary to evaluate output produced by the prototype in the context of the organisation being investigated.

6.4. Recommendations

This section identifies opportunities for future research. Such research is recommended to address the limitations of the current study and to justify further the applicability of the conceptual model developed in this study.

6.4.1. Recommendations for further research

There are a few issues that arise from this study which may provide opportunities for further research. Data collected for this study is limited to a single case. This is mainly due to organisations not being prepared to provide access to their transaction data. Future research could extend this study by replication in other organisations locally and internationally to test whether the same findings are observed or not.

This study is limited to billing fraud schemes involving shell companies and non-accomplice vendors within accounts payable thereby restricting generalisability of results. The ACFE fraud tree (ACFE 2010) provides a detailed classification of occupational fraud. Extending the focus of the catalogue of fraud symptoms and

fraud detection algorithms developed in this study to include other fraud schemes will extend identification of potential threats or frauds.

A limited number of experts were involved in validation of the prototype. Participation was voluntary. To reduce possible bias from feedback, future research could extend this study by inviting and selecting experts from a wider group. A call for experts could be issued and the recruiting process could possibly be managed by an external organisation. Applications could be selected on the basis of criteria developed for the study. Additionally, a single panel session of 60 minutes was conducted in this study. Future research could extend this to multiple sessions, incorporating feedback from each of these sessions into the design of the conceptual model and prototype. Further independent reviews could also be commissioned to offer more diverse feedback and validation comments.

The prototype may incorrectly identify anomalous activities due to Type I and Type II errors. Future research could incorporate organisational profiles within the prototype design. These profiles could be developed in conjunction multiple participating organisations and be based on their implementation of segregation of duties and internal control policies. A database of diverse profiles could be developed. By applying the relevant profile prior to performing data analytics, a reduction on Type I or Type II errors may possibly be observed. Further enhancements to the prototype to reduce potential errors and improve potential detection are discussed in Section 6.4.2.

An effective fraud risk management process has three objectives: prevention, detection and response (Albrecht et al. 2009 ; Romney and Steinbart 2009 ; Wells 2008 ; KPMG 2006). The focus of this study is on fraud detection. An outcome identified in this study is that proactive fraud detection enables organisations to quickly and efficiently identify frauds that circumvent preventative measures and take appropriate corrective action thereby reducing the propensity for losses associated with future fraud. Expert panel members confirmed this outcome by commenting that the prototype could potentially reduce the propensity for future fraud due to early detection. This outcome is embodied in the conceptual model (Figure 3.3). Future research could collect and examine empirical data from organisations that implement proactive fraud detection software to determine the actual effect it has on occurrence of fraud.

This study develops a *Fraud Risk Index* that determines the propensity for potential fraud occurring in an organisation being investigated. The value of the index is calculated from variables shown in Table 4.6. The following formula is used to calculate the index.

$\text{FRAUD RISK INDEX} = \text{riskyusers} + \text{vendsharingbank} + \text{vendmultibank} + \text{vendbankchanges} + \text{beninv} + \text{benpmt} + \text{rinv} + \text{rpmt}$
--

Weights assigned to variables are based on estimates of the importance of individual variables and by defining ranges within variables. By replicating this study in multiple organisations, future research could collect empirical evidence on the accuracy of the index and use this evidence to adjust values of weights. This may

improve accuracy of prediction. The calculated result is a value out of **10**. The index is intended to create awareness of the potential of fraud occurring in an organisation; however, individual circumstances within an organisation must be considered when basing decisions on this value.

Fraud, by its very nature, does not lend itself to being measured very accurately due to its clandestine nature (ACFE 2010). Any measurement of fraud is at best an estimate. Although this study recognised underlying factors that motivate an individual to commit fraud, it did not examine empirical evidence to confirm the relationship between these factors and perpetration of fraud. Future research could examine these factors to determine behavioural aspects of an organisation's employees and its influence on fraud. It is these underlying factors that create opportunities for committing fraud in organisations.

6.4.2. Recommendations for extensions to prototype

The prototype developed in this research is intended to address the primary research question by providing evidence that *"a generalised model for proactive detection of potential fraud in enterprise systems can be developed"*. Such a prototype is meant to demonstrate that the *"concept of proactive fraud detection"* is feasible in practice. It is a limited version meant for showcasing the concept and for testing purposes only.

Audit trails maintained in a SAP enterprise system form the basis of the fraud detection process. Accordingly, the integrity of these audit trails is of vital importance in assessing usefulness and accuracy of the detection process. System

and security administrators have the capability to create and/or maintain users. They may create fake users and act in their name. Users identified through the fraud detection process must be investigated by an auditor to determine whether they are real or not.

Fraudsters may be aware of the implementation of a fraud detection system in an organisation and hence be cautious of having their activities monitored. Consequently, there is the threat of collusion between users. In this scenario no one user performs all required tasks to perpetrate a fraud, making it difficult to identify potential fraud. They may also use a combination of fraud perpetration methods. The prototype is capable of identifying multiple users performing transactions on the same vendor; however, it does not conclusively detect collusion. The obligation is on an auditor to further investigate activities of users identified through this process.

Every installation of SAP is unique due to its modular architecture (refer to Section 2.9). The financial and controlling sub-system (FICO) is common across all SAP installations. The prototype makes no assumption about individual SAP installations. It relies exclusively on transaction data obtained from the FICO sub-system. Extending the prototype to operate on additional sub-systems may improve effectiveness.

The scope of fraud detection may be broadened to incorporate other sub-systems including accounts receivable (AR), general ledger (GL), sales and distribution (SD), materials management (MM) and human resources (HR). This requires extending the database of critical combinations and known fraud symptoms to incorporate

transaction codes and rules pertinent to these areas. A desirable extension would be to identify possible collusion between users across functional areas in an organisation. This could be achieved by tracking associations between users and vendors in a Collusion database. Additionally, incorporating data from the human resources (HR) sub-system would facilitate matching of vendor details with employee details. Unless employees are paid as vendors, there should be no match. Furthermore, vendor address details may be validated with online business address databases to ensure legitimacy.

A concept relevant to the analysis of user activity is the 'role' of a user in an organisation. The prototype could be extended to provide an optional *user roles* feature. The 'role' of each user (such as accounts payable clerk, payroll clerk, SAP support, accounts receivable clerk) could be maintained in a Settings database. Reporting facilities could be made available which focus on differences in profiles among users performing the same role in an organisation.

Masquerading occurs when a fraudster logs in to an enterprise system as a target user, using their user identification and password. The fraudster may act in the name of the target user, and perform vendor maintenance or accounting transactions. Where the target user has elevated privileges, the intruder may use system utilities to modify the security characteristics of the system, such as adding new users, changing passwords and inflating privileges. Browsing refers to attempts by authorised users to obtain private information (such as user-ids) to assist in the above methods, or to perform unauthorised functions, such as accessing sensitive transaction data, changing user privileges, printing or displaying transaction and vendor data (Best et

al. 2004). Anomalous computer workstation usage may be indicative of masquerading or browsing attempts described above. This may be indicative of potential fraud. The prototype could be extended to maintain a Settings database defining sets of incompatible workstations and to report on users who access multiple workstations in a specified period.

Data extraction is a manual process requiring a user with elevated SAP privileges to perform an extraction. A desirable extension to the prototype would be to automate the data extraction process. This may be achieved by developing ABAP programs that run within a SAP system or by using third-party applications such as Direct Link for SAP from ACL. Automating data selection and extraction will enhance data access, analysis, and reporting capabilities of the prototype.

It is envisaged that an auditor may wish to investigate a group of users who have been associated with anomalous activity during a given period. Apart from monitoring their activities during the period under review, the auditor may wish to look back at actions during other earlier periods. The prototype does provide limited archiving of audit trails for a period of one year. A desirable extension to the system would be to provide an optional *archiving and retrieval* feature. Such a facility would require an auditor to specify the retention period for audit trails and provide the ability to retrieve and analyse records for a specified time period.

Figure 6.1 illustrates an extended model of the prototype. The proposed optional features in such a system are:

- i) access to online business directories for vendor address validation;

- ii) extended critical combinations and known fraud symptoms databases incorporating additional t-codes and rule-sets;
- iii) multiple workstation monitoring;
- iv) user role monitoring;
- v) automated data extraction; and
- vi) archiving and retrieval of audit trails.

From the preceding discussion it can be noted that several research opportunities exist for extending the prototype. Research is also recommended to justify further the applicability of audit trail analysis for proactive fraud detection, to assess the potential application of data visualisation to expose patterns of activity, and to gauge the potential social impact of monitoring user activity. These opportunities are discussed below.

Visualisation for fraud detection

The eye processes information more efficiently when presented as images as opposed to textual information. As our instincts develop over time so does our ability to process complex concepts through visual identification. By representing information spatially and with images, humans are able to grasp its meaning, to group similar ideas and to connect it with prior knowledge effortlessly. Using illustrations or diagrams to represent large amounts of information facilitates easier understanding and helps reveal patterns and relationships (Pashler et al. 2008).

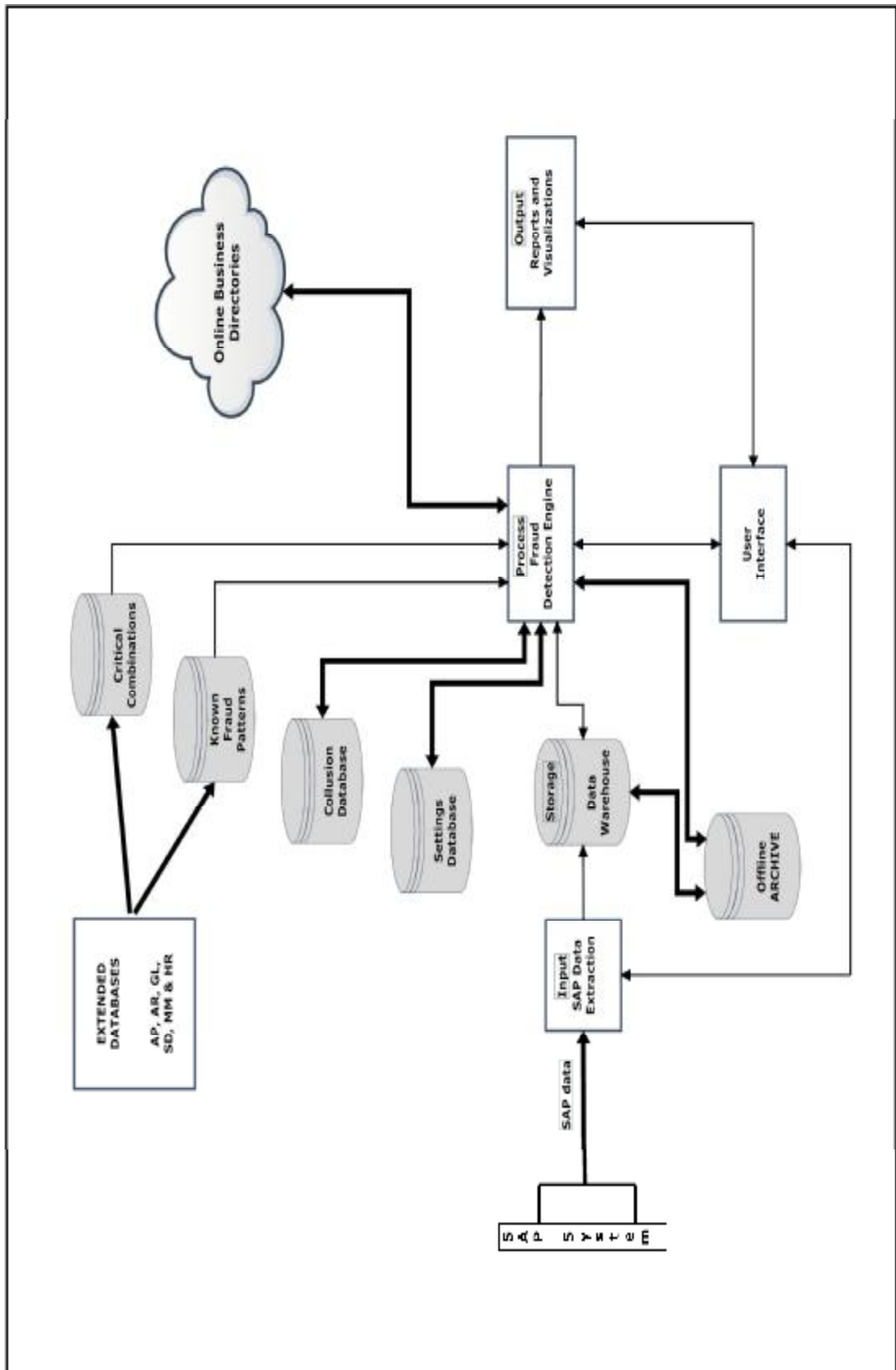


Figure 6.1: Model of extended prototype

Note: Extensions recommended are shaded

Visualisation facilitates investigation of large data sets and makes it possible to find new trends, patterns and threats that would otherwise take hours or days using conventional techniques. While the size of transaction data-sets continues to grow exponentially, tools and techniques to visualise and discover what is in the data has not significantly changed.

The complex nature of fraud and other 'white-collar' crimes requires visualisation tools that can view and leverage the enormous amount of information being generated in this digital age. Thousands of transactions daily generate thousands of lines of data in an enterprise system. Hidden among these gigabytes of data may possibly be fraudulent transactions that are near impossible to detect. Forensic analysts and auditors need as much assistance as they can get to find these threats. Previous attempts at visualisation have yielded limited success having relied mainly on finding one-to-one relationships between entities (Marane 2008).

This study has demonstrated some, albeit limited, use of visualisation to detect fraud. Feedback from expert panel members indicated that visualisations were especially useful in promptly identifying anomalies. Further research is recommended on the potential use of visualisation for fraud detection.

Analysis of EFTPOS transactions

EFTPOS transactions are a special type of accounting transaction. These are transactions initiated by customers of financial institutions to deposit or withdraw cash from savings or cheque accounts, to transfer funds between accounts or to make payments to suppliers of goods or services. They are conducted using an automatic

teller machine (ATM) linked to the financial institution's computer system or another EFTPOS terminal on a supplier's premises. Authorisation for these transactions is established by the customer having an appropriate transaction card in their possession and knowing the personal identification number (PIN). While the EFTPOS system is used heavily throughout the financial community, there is widespread concern regarding the extent of transaction card fraud. Theft of cards or use of fake cards, coupled with PIN guessing techniques, is of major concern to financial institutions. In 2010, fraud perpetrated on Australian issued payment instruments exceeded \$211 million (ACPA 2011). EFTPOS scams are regularly reported in the press (ABCNews 2010 ; WAToday 2009).

Social implications of audit trail analysis

Organisations are increasingly using technology with policy to monitor and manage employees' productivity and to minimise litigation, security and other risks (AMA 2007). Whilst the IT department may most frequently be responsible for the monitoring, their main concern is that employees are not abusing resources such as e-mail or downloading large files from the Internet (D'Agostino 2006). Managers, on the other hand usually want to monitor what their staff are doing. That kind of monitoring may create a stressful work environment, which may lead to higher staff turnover, job dissatisfaction and erosion of trust between employee and employer.

Employers justify electronic monitoring as promoting business interests. Yet the practice raises concerns from all areas of society, business organisations, employee interest groups, lawyers, and civil libertarians. Each group cites economic, legal and ethical rationales in support of their position. No argument, however, is conclusive

and each raises important managerial and moral issues (Young 2011 ; Riedy and Wen 2010 ; Nancherla 2008).

A primary concern may be the possible impact on individuals who are wrongly accused by a software tool that is merely applying rules and procedures to identify patterns of activity. The limitations of the system and its output may be poorly understood by an auditor. The targeted user may have limited opportunities to prove their innocence. Safeguards are needed to protect the rights of individuals in such cases. In the case of the prototype, proper education of auditors and users are crucial for the effective use and interpretation of its output. Further research is recommended on the potential social impact of audit trail analysis.

6.5. Conclusion

Australia has an estimated \$3 billion per year financial fraud problem that continues to worsen (Standards Australia 2008). Enhancing the ability of organisations to detect potential fraud may have a positive impact on the economy. An effective model that facilitates proactive detection of potential fraud may potentially save costs and reduce the propensity of future fraud by early detection of suspicious user activities.

Enterprise systems generate hundreds of thousands to millions of transactions annually. While most of these are legal and routine transactions, a small number may be fraudulent. The enormous amount of generated transactions makes it difficult to find these few instances among legitimate transactions. Without the availability of proactive fraud detection tools, investigating suspicious activities becomes

overwhelming. The prototype developed in this study may assist auditors in detecting potential fraud by retrospective monitoring of enterprise system audit trails and reporting these using an intuitive visual interface. Violations in segregation of duties may be identified by analysing audit trails for critical combinations of user activities. Potentially fraudulent transactions may be identified by investigating user activities that violate segregation of duties, match known fraud symptoms, or appear otherwise anomalous.

The prototype may be a valuable tool to organisations with large-scale implementations of enterprise systems as it automates routine data analytics thereby improving auditor productivity and reducing time taken to identify potential fraud.

This research has demonstrated the *feasibility of implementing proactive detection of potential fraud in enterprise systems*.

ooOoo

LIST OF REFERENCES

- ABCNews (2010) EFTPOS scam costs Australians \$80 million, <http://www.abc.net.au/news/2010-07-15/eftpos-scam-costs-australians-80m/906008>. Accessed: 09/01/2012
- Abu-Musa, A. A. (2007) Evaluating the security controls of CAIS in developing countries: An examination of current research. *Information Management & Computer Security*, Emerald, 15 (1), 46-63.
- ACFE (2010) Report to the Nation on Occupational Fraud and Abuse, <http://www.acfe.com/rtn>. Accessed: 6/10/2010
- ACPA (2011) Fraud Perpetrated on Australian Issued Payment Instruments 1 January 2010 - 31 December 2010 (Revised December 2011), http://www.apca.com.au/Public/apca01_live.nsf/WebPageDisplay/FraudStats_2010B_Summary?openDocument. Accessed: 09/01/2012
- Adams, M. B. (1994) Agency Theory and the Internal Audit. *Managerial Auditing Journal*, 9 (8), 8-12.
- Adrion, W. R., Branstad, M. A. & Cherniavsky, J. C. (1982) Validation, Verification, and Testing of Computer Software. *ACM Comput. Surv.*, 14 (2), 159-192.
- Albrecht, W. S., Albrecht, C. & Albrecht, C. C. (2008) Current Trends in Fraud and its Detection. *Information Security Journal: A Global Perspective*, Taylor & Francis Ltd, 17 (1), 2-12.
- Albrecht, W. S., Albrecht, C. C. & Albrecht, C. D. (2009) *Fraud Examination*. 3rd Ed., Thomson/South-Western.
- Alles, M., Brennan, G., Kogan, A. & Vasarhelyi, M. A. (2006) Continuous monitoring of business process controls: A pilot implementation of a continuous auditing system at Siemens. *International Journal of Accounting Information Systems*, 7 (2), 137-161.
- Alles, M. G., Kogan, A. & Vasarhelyi, M. A. (2002) Feasibility and Economics of Continuous Assurance. *AUDITING: A Journal of Practice & Theory*, 21 (1).
- Alles, M. G., Kogan, A. & Vasarhelyi, M. A. (2004) Restoring auditor credibility: tertiary monitoring and logging of continuous assurance systems. *International Journal of Accounting Information Systems*, 5 (2), 183-202.
- Alles, M. G., Kogan, A. & Vasarhelyi, M. A. (2008) Putting Continuous Auditing Theory into Practice: Lessons from Two Pilot Implementations. *Journal of Information Systems*, American Accounting Association, 22 (2), 195-214.

- AMA (2007) 2007 Electronic Monitoring and Surveillance Survey, *AMA/ePolicy Institute Research*. Accessed: 28/12/2011
- ARC (2011) ARC Research Priority 4: Safeguarding Australia, *Australian Government*, <http://www.research.swinburne.edu.au/grants-contracts/funding/arc/research-priority-4.html>. Accessed: 08/06/2012
- Arens, A. A., Best, P., Shailer, G., Fiedler, B., Elder, R., J; & Beasley, M. (2007) *Auditing and Assurance Services in Australia: An Integrated Approach*. 7th, Prentice Hall.
- ASB (2002) Statement on Auditing Standards No. 99: Consideration of Fraud in a Financial Statement. American Institute of Certified Public Accountants.
- Asur, S. & Hufnagel, S. (1993) Taxonomy of rapid-prototyping methods and tools. *Rapid System Prototyping, 1993. Shortening the Path from Specification to Prototype. Proceedings., Fourth International Workshop on*.
- AuditNet (2011) Study Shows Auditors Slow to Adopt Hi-Tech Fraud Detection Strategies, *Bay Street Group LLC*, http://cpatrendlines.com/2011/12/05/study-shows-auditors-slow-to-adopt-hi-tech-fraud-detection-strategies/?utm_source=dlvr.it&utm_medium=twitter. Accessed: 17/01/2012
- Baran, P. A. & Sweezy, P. M. (1970) *Monopoly Capital: An essay on the American Order*. Harmondsworth, England, Penguin Books.
- Beattie, A. (2009) The Pioneers of Financial Fraud, *Investopedia*, <http://www.investopedia.com/articles/financial-theory/09/history-of-fraud.asp#axzz1jOJOIW6K>. Accessed: 14/01/2012
- Berle, A. A. & Means, G. C. (1932) *The modern corporation and private property*. Harcourt, Brace & World, Inc.
- Best, P. J. (2000) SAP R/3 Audit Trail Analysis. *Sapphire 2000. 4th Annual SAP Asia Pacific Institute of Higher Learning Forum*. Brisbane, Australia.
- Best, P. J. (2005) Audit Trail Analysis For Fraud Control With SAP R/3. *Oceania Computer Audit, Control and Security Conference (CACCS) 2005 Conference*. Perth, Australia.
- Best, P. J. (2008) SAP - Accounts Payable. *ACC3101 - Accounting Information Systems*. USQ.
- Best, P. J., Mohay, G. & Anderson, A. (2004) Machine-Independent Audit Trail Analysis – A Decision Support Tool for Continuous Audit Assurance. . *International Journal of Intelligent Systems in Accounting, Finance & Management* 12 (2), 85-102.
- Best, P. J., Rikhardson, P. & Toleman, M. (2009) Continuous Fraud Detection in Enterprise Systems through Audit Trail Analysis. *Journal of Digital*

Forensics, Security and Law, Association of Digital Forensics, Security and Law, 4 (1).

Bologna, J. (1992) thinking Like a Thief. *The Internal Auditor*, ABI/INFORM Global, 49 (4), 30-33.

Bond, G. (2011) Security firm boss pleads guilty to \$1.4million fraud. Auckland, NZ, *The National Business Review*.

BOS (2009) Benefits of Using SAP for Your Business, http://www.bos.com.np/index.php?option=com_content&view=article&id=61:benefits-of-using-sap-for-your-business&catid=34:articles&Itemid=72. Accessed: 08/11/2010

Bouguettaya, A., Malik, Z., Rezgui, A. & Korff, L. (2006) A Scalable Middleware for Web Databases. *Journal of Database Management*, 17 (4), 20-46.

Broady, D. V. & Roland, H. A. (2008) *SAP GRC For Dummies*. John Wiley and Sons.

Budde, R. & Zullighoven, H. (1990) Prototyping revisited. *CompEuro '90. Proceedings of the 1990 IEEE International Conference on Computer Systems and Software Engineering*.

Casabona, P. A. & Grego, M. J. (2003) SAS 99 - Consideration of Fraud in a Financial Statement Audit: A Revision of Statement on Auditing Standards 82. *Review of Business*, St. John's University, 24 (2), 16.

Chadwick, B. A., Bahr, H. M. & Albrecht, S. L. (1984) *Social Science Research Methods*. Englewood Cliffs, New Jersey, Prentice-Hall.

Chang, R., Ghoniem, M., Kosara, R., Ribarsky, W., Jing, Y., Suma, E., Ziemkiewicz, C., Kern, D. & Sudjianto, A. (2007) WireVis: Visualization of Categorical, Time-Varying Data From Financial Transactions. *Visual Analytics Science and Technology, 2007. VAST 2007. IEEE Symposium on*.

CMU (2011) Companies using SAP?, *Central Michigan University*, http://sapua.cba.cmich.edu/sap_usersDB/. Accessed: 19/12/2012

Coase, R. H. (1937) The Nature of the Firm. *Economica*, 4 (16), 386-405.

Coderre, D. & Warner, P. D. (1999) Computer-Assisted Techniques for Fraud Detection. *CPA Journal*, New York State Society of CPAs, 69 (8), 57.

Coderre, D. G. (2005) *Global Technology Audit Guide. Continuous Auditing: Implications for Assurance, Monitoring, and Risk Assessment*. IN IIA (Ed.). Florida, IIA.

Coenen, T. (2008) *Essentials of Corporate Fraud*. John Wiley and Sons.

- Cojocariu, A., Munteanu, A. & Sofran, O. (2005) Verification, Validation and Evaluation of Expert Systems in Order to Develop a Safe Support in the Process of Decision Making, *Computational Economics, EconWPA*, <http://ideas.repec.org/p/wpa/wuwpc0/0510002.html>. Accessed: 10/11/2011
- Coleman, K. (2008) Separation of Duties and IT Security, *CSO Security and Risk*, <http://www.csoonline.com/article/446017/separation-of-duties-and-it-security>. Accessed: 08/06/2012
- Collins (2000) Collion English Dictionary, *Harper Collion* <http://dictionary.reverso.net/english-definition/expert%20panel>. Accessed: 17/12/2011
- ComLaw (2011) Commonwealth Fraud Control Guidelines - F2011L00511, <http://www.comlaw.gov.au/Details/F2011L00511/Html/Text>. Accessed: 18/01/2012
- COSO (1992) Internal Control - Integrated Framework, *Committee of Sponsoring Organisations of the Treadway Commission*, <http://www.coso.org/ic-integratedframework-summary.htm>. Accessed: 12/06/2012
- CPA (2009) Clarity Standard ASA 320. Materiality in Planning and PERforming and Audit., *CPA Australia*, <http://www.cpaaustralia.com.au/cps/rde/xbcr/cpa-site/ASA320-materiality-Business-FactSheets.pdf>. Accessed: 21/01/2012
- Cressey, D. R. (1950) The Criminal Violation of Financial Trust. *American Sociological Review*, Sage Publications Inc., 15 (6), 738-743.
- Cressey, D. R. (1953) *Other people's money; a study of the social psychology of embezzlement*. Other people's money; a study of the social psychology of embezzlement. New York, NY US, Free Press.
- Creswell, J. W. (2005) *Educational Research: Planning, Conducting, and Evaluating Quantitative and Qualitative Research* New Jersey, Pearson Education, Inc.
- D'Agostino, D. (2006) IT Spying on the Rise. *CIO Insight*, Ziff Davis Enterprise, (67), 30-30.
- Daigle, R. J. & Lampe, J. C. (2004) The impact of the risk of consequence on the relative demand for continuous online assurance. *International Journal of Accounting Information Systems*, 5 (3), 313-340.
- Davis, A. M. (1992) Operational prototyping: a new development approach. *Software, IEEE*, 9 (5), 70-78.
- De Vaus, D. A. (2001) *Research design in social research.*, London, SAGE.
- Debreceeny, R. S. & Gray, G. L. (2010) Data mining journal entries for fraud detection: An exploratory study. *International Journal of Accounting Information Systems*, 11 (3), 157-181.

- Debreceeny, R. S., Gray, G. L., Jun-Jin Ng, J., Siow-Ping Lee, K. & Yau, W.-F. (2005) Embedded Audit Modules in Enterprise Resource Planning Systems: Implementation and Functionality. *Journal of Information Systems*, American Accounting Association, 19 (2), 7-27.
- Demsetz, H. (1983) The Structure of Ownership and the Theory of the Firm. *Journal of Law and Economics*, 26 (2), 375-390.
- Denning, D. E. (1987) An Intrusion-Detection Model. *Software Engineering, IEEE Transactions on*, SE-13 (2), 222-232.
- Deshmukh, A. (2006) *Digital Accounting: The Effects of the Internet and ERP on Accounting*. IGI Global.
- Du, H. & Roohani, S. (2007) Meeting Challenges and Expectations of Continuous Auditing in the Context of Independent Audits of Financial Statements. *International Journal of Auditing*, 11 (2), 133-146.
- EC (2011) European Commission. Regional Policy. Source Book: Method and techniques. Expert Panels., *European Commission*, http://ec.europa.eu/regional_policy/sources/docgener/evaluation/evalsed/sourcebooks/method_techniques/collecting_information/expert_panels/index_en.htm. Accessed: 19/12/2011
- Edge, M. E. & Falcone Sampaio, P. R. (2009) A survey of signature based methods for financial fraud detection. *Computers & Security*, 28 (6), 381-394.
- Ehie, I. C. & Madsen, M. (2005) Identifying critical issues in enterprise resource planning (ERP) implementation. *Computers in Industry*, 56 (6), 545-557.
- Ernst & Young (2002) Preparing for Internal Control Reporting: A Guide for Management's Assessment under Section 404 of the SarbanesOxley Act. New York, NY, Ernst and Young LLP.
- Fama, E. F. & Jensen, M. C. (1983) Separation of Ownership and Control. *Journal of Law & Economics*, XXVI.
- Farnsworth, J. & Boon, B. (2010) Analysing group dynamics within the focus group. *Qualitative Research*, 10 (5), 605-624.
- Feagin, J., Orum, A. & Sjoberg, G. (1991) *A case for case study.*, Chapel Hill, NC, University of North Carolina Press.
- Fetaji, B. (2011) Development and Analyses of Dynamical Visualization Process Tool in Run Time and its Usability Evaluation. *TTEM- Technics Technologies Education Management*, TTEM-Technics Technologies Education Management, 6 (2), 447-454.
- Flowerday, S. & von Solms, R. (2005) Continuous auditing: verifying information integrity and providing assurances for financial reports. *Computer Fraud & Security*, 2005 (7), 12-16.

- Fuller, S. J. (2002) Expert Panels and Customer Group Sessions, *Informedix Marketing Research, Inc*, <http://www.google.com.au/url?sa=t&rct=j&q=focus%20group%20expert%20panel&source=web&cd=1&ved=0CDkQFjAA&url=http%3A%2F%2Fwww.informedixmr.com%2Fresources%2Fpdfs%2FExpert%2520Panels%2520and%2520Customer%2520Group%2520Sessions.pdf&ei=CN3rTp3-JYqSiQfNpaCGBw&usg=AFQjCNFPwGnzqlLeeR3V1wYET9j7Ca3QzA&ad=rja>. Accessed: 17/12/2011
- Gartner (2010) Gartner Says Worldwide Business Intelligence, Analytics and Performance Management Software Market Grew 4 Percent in 2009, *Gartner, Inc*, <http://www.gartner.com/it/page.jsp?id=1357514>. Accessed: 27/10/2010
- Gill, W. (2009) Fighting Fraud with Advanced Analytics. *Canadian Underwriter*, Business Information Group, 76 (9), 28-32.
- Goode, S. & Lacey, D. (2011) Detecting complex account fraud in the enterprise: The role of technical and non-technical controls. *Decision Support Systems*, 50 (4), 702-714.
- Gopalakrishna, R. (2000) Audit Trails, *Purdue University*, <http://homes.cerias.purdue.edu/~rgk/at.html>. Accessed: 12 November 2010
- GraphViz (2010) Graphviz - Graph Visualization Software, <http://www.graphviz.org/About.php>. Accessed: 21/12/2011
- Green, C. D. (2000) Classics in the History of Psychology, *York University*, <http://psychclassics.yorku.ca/Maslow/motivation.htm>. Accessed: 11 March
- Greene, C. L. (2003a) Audit Those Vendors, *The White Paper, McGovern & Greene*, http://www.mcgovernandgreene.com/archives/archive_articles/Craig_Greene_Archives/audit_vendors.html. Accessed: 21/09/2010
- Greene, C. L. (2003b) Focus on Employee Frauds - Purchasing Frauds, *McGovern & Greene*, http://www.mcgovernandgreene.com/archives/archive_articles/Craig_Greene_Archives/Focus-Employee_Frauds-Purch.html. Accessed: 29/09/2010
- Groomer, S. M. & Murthy, U. S. (1989) Continuous Auditing of Database Applications: An Embedded Audit Module Approach. *Journal of Information Systems*, American Accounting Association, 3 (2), 53.
- Groomer, S. M. & Murthy, U. S. (2003) Monitoring High Volume On-line Transaction Processing Systems Using a Continuous Sampling Approach. *International Journal of Auditing*, 7 (1), 3-19.
- Hernandez, J. A. (2002) *Roadmap to mySAP.com*. Premier Press.
- Hernandez, J. A., Keogh, J. & Martinez, F. (2006) *SAP R/3 Handbook*. 3rd ed., McGraw Hill/Osborne.

- Herzberg, F., Mausner, B. & Snyderman, B. B. (1959) *The Motivation to Work*. New York, John Wiley.
- Hirao, J. (2009) *SAP Security Configuration and Deployment: The IT Administrator's Guide to Best Practices*. Burlington, MA, Syngress Publishing.
- Hunton, J. E., Wright, A. M. & Wright, S. (2004) Continuous Reporting and Continuous Assurance: Opportunities for Behavioral Accounting Research. *Journal of Emerging Technologies in Accounting*, American Accounting Association, 1, 91-102.
- IEEE (2004) Guide to the Software Engineering Body of Knowledge (SWEBOK), *IEEE Computer Society*, <http://www.computer.org/portal/web/swebok/html/ch11>. Accessed: 14/11/2011
- ITGI (2006) *IT Objectives for Sarbanes-Oxley*. 2nd ed. Rolling Meadows IL, IT Governance Institute.
- Jensen, M. C. & Meckling, W. H. (1976) Theory of the Firm: Managerial Behaviour, Agency Costs and Ownership Structure. *Journal of Financial Economics*, 3 (4), 305-360.
- Kamhawi, E. M. (2008) Enterprise resource-planning systems adoption in Bahrain: motives, benefits, and barriers. *Journal of Enterprise Information Management*, Emerald Group Publishing, 21 (3), 310-334.
- Koch, C. & Wailgum, T. (2008) ERP Definition and Solutions, *CXO Media*, http://www.cio.com/article/40323/ERP_Definition_and_Solutions. Accessed: 29/10/2010
- Kogan, A., Sudit, E. F. & Vasarhelyi, M. A. (1999) Continuous Online Auditing: A Program of Research. *Journal of Information Systems*, American Accounting Association, 13 (2), 87.
- Kotb, A. & Roberts, C. (2011) The Impact of E-Business on the Audit Process: An Investigation of the Factors Leading to Change. *International Journal of Auditing*, 15 (2), 150-175.
- KPMG (2004) Fraud Survey 2004, *KPMG*, <http://www.kpmg.com>. Accessed: 16/04/2007
- KPMG (2006) Fraud Risk Management, *KPMG Forensic*, <http://www.kpmg.com>. Accessed: 18/01/2010
- KPMG (2007) Profile of a Fraudster Survey 2007, <http://www.kpmg.com>. Accessed: 18/03/2009
- KPMG (2008) Fraud Survey 2008, *KPMG*, <http://www.kpmg.com>. Accessed: 03/11/2009

- KPMG (2009) Fraud Survey 2009, *KPMG*, <http://www.kpmg.com>. Accessed: 18/01/2010
- KPMG (2010) Fraud and Misconduct Survey 2010, *KPMG*, www.kpmg.com. Accessed: 19/01/2012
- Krakowiak, S. (2007) What is Middleware, *ObjectWeb Open Source Middleware*, <http://middleware.objectweb.org/>. Accessed: 18/01/2012
- Krueger, R. A. & Casey, M. A. (2000) *Focus Groups: A practical guide for applied research*. Newbury Park, Sage
- Kuhn Jr, J. R. & Sutton, S. G. (2010) Continuous Auditing in ERP System Environments: The Current State and Future Directions. *Journal of Information Systems*, American Accounting Association, 24 (1), 91-112.
- Kuhn, J. R. & Sutton, S. G. (2006) Learning from WorldCom: Implications for Fraud Detection through Continuous Assurance. *Journal of Emerging Technologies in Accounting*, 3 (1), 61-80.
- Lager, M. & Tsai, J. (2008) SAP Retains Market-Share Lead in CRM. *Customer Relationship Management*, Insight, (October 2008), 17-18.
- Lanza, R. B. (2003) Proactively Detecting Occupational Fraud Using Computer Audit Reports. Florida, The IIA Research Foundation.
- Lanza, R. B. (2007) Auditing Vendor Accounts for Fraud or at least some Cash Recovery. *Fraud Magazine*. September/October 2007 ed. Austin, ACFE.
- Li, N., Tripunitara, M. V. & Bizri, Z. (2007) On mutually exclusive roles and separation-of-duty. *ACM Trans. Inf. Syst. Secur.*, 10 (2), 5.
- Liang, L. Y. & Miranda, R. (2001) Dashboards and Scorecards: Executive Information Systems for the Public Sector. *Government Finance Review*, CBS Interactive Business Network.
- Little, A. & Best, P. J. (2003) A framework for separation of duties in an SAP R/3 environment *Managerial Auditing Journal* 18 (5), 419-430.
- Luqi, L. & Steigerwald, R. (1992) Rapid software prototyping. *System Sciences, 1992. Proceedings of the Twenty-Fifth Hawaii International Conference on*.
- Marane, A. (2008) Visual Analysis of Large Datasets, <http://linkanalysisnow.com/2011/07/visual-analysis-of-large-datasets.html>. Accessed: 09/01/2012
- Maslow, A. H. (1943) A Theory of Human Motivation. *Psychological Review*, 50 (4), 370-396.
- Maslow, A. H. (1954) *Motivation and Personality*. New York, Harper.

- Matthews, D. (2006) FROM TICKING TO CLICKING: CHANGES IN AUDITING TECHNIQUES IN BRITAIN FROM THE 19th CENTURY TO THE PRESENT. *Accounting Historians Journal*, Academy of Accounting Historians, 33 (2), 63-102.
- Morgan, D. L. (1997) *Focus Grpups As Qualitative Research*. Qualitative Research Methods Series. 2nd., Thousand Oaks, California, Sage Publications Inc.
- Moy, L. (2008) A Methodology Update on Focus Groups, Expert Panels, and other Small Group Methods. Portland, OR, US Government Accountability Office.
- Murphy, P. & Dacin, M. (2011) Psychological Pathways to Fraud: Understanding and Preventing Fraud in Organizations. *Journal of Business Ethics*, Springer Science & Business Media B.V., 101 (4), 601-618.
- Murthy, U. S. & Groomer, S. M. (2004) A continuous auditing web services model for XML-based accounting systems. *International Journal of Accounting Information Systems*, 5 (2), 139-163.
- Musaji, Y. F. (2002) *Integrated Auditing of ERP Systems*. John Wiley & Sons.
- Nancherla, A. (2008) SURVEILLANCE: Increases in Workplace. *T+D*, American Society for Training & Development, 62 (5), 12-12.
- Narayan, V. (2008) *Financial Accounting (FI). SAP FI/CO Questions and Answers.*, Sudbury, Infinity Science Press.
- Naumann, J. D. & Jenkins, A. M. (1982) Prototyping: The New Paradigm for Systems Development. *MIS Quarterly*, MIS Quarterly & The Society for Information Management, 6 (3), 29-44.
- Nigrini, M. J. (2011) *Forensic Analytics. Methods and Techniques for Forensic Accounting Investigations*. New Jersey, John Wiley & Sons
- NIST (2003) NIST/SEMATECH e-Handbook of Statistical Methods. IN CROARKIN, C. & TOBIAS, P. (Eds.). US Department of Commerce.
- NIST (2005) An Introduction to Computer Security: The NIST Handbook. *Special Publication 800-12*. US Department of Commerce.
- Norris, G., Hurley, J. R., Hartley, K. M., Dunleavy, J. R. & Balls, J. D. (2000) *E-Business and ERP: Transforming the Enterprise*. John Wiley & Sons.
- Norris, G., Wright, I., Hurley, J. R., Dunleavy, J. & Gibson, A. (1998) *SAP: An Executive's Comprehensive Guide*. Wiley.
- O'Gara, J. D. (2004) *Corporate Fraud Case Studies in Detection and Prevention*. Wiley.
- Oliver, M. S. (1999) *Information Technology Research. A Practical Guide.*, Johannesburg, UNISA Press.

- Oxford (2012) Oxford Dictionaries, *Oxford University Press*, <http://oxforddictionaries.com>. Accessed: 09/01/2012
- Padhi, S., N (2010) *SAP ERP Financials and FICO Handbook*. Sudbury, Jones and Bartlett.
- Panorama (2011) ERP Market Share and Vendor Evaluation, *Panorama Consulting*, <http://whatiserp.net/erp-report/erp-market-share-and-vendor-evaluation-2011/>. Accessed: 27/11/2011
- Parker, A. & Tritter, J. (2006) Focus group method and methodology: current practice and recent debate. *International Journal of Research & Method in Education*, 29 (1), 23-37.
- Pashler, H., McDaniel, M., Rohrer, D. & Bjork, R. (2008) Learning Styles: Concepts and Evidence. *Psychological Science in the Public Interest (Wiley-Blackwell)*, Wiley-Blackwell, 9 (3), 105-119.
- Patton, M. Q. (1990) *Qualitative evaluation and research methods (2nd ed.)*. Qualitative evaluation and research methods (2nd ed.). Thousand Oaks, CA US, Sage Publications, Inc.
- Perry, C., Riege, A. & Brown, L. (1999) Realism's role among scientific paradigms in marketing research. *Irish Marketing Review*, The Marketing Institute, 12 (2), 16-23.
- Plummer-D'Amato, P. (2008) Focus group methodology part 1: considerations for design. *International Journal of Therapy & Rehabilitation*, 15 (2), 69-73.
- Potla, L. (2003) Detecting Accounts Payable Abuse Through Continuous Auditing. *ITAudit*.
- Presley, A. (2006) ERP investment analysis using the strategic alignment model. *Management Research News*, 29 (5), 273-284.
- PricewaterhouseCoopers (2002) Strategies for Meeting New Internal Control Reporting Challenges: A White Paper—The Sarbanes-Oxley Act of 2002. New York, NY, PricewaterhouseCoopers.
- PwC (2009) The Global Economic Crime Survey. Economic crime in a downturn. November 2009, *Pricewaterhouse Coopers*, <http://www.pwc.com/gx/en/economic-crime-survey>. Accessed: 08/02/2010
- Remenyi, D. (1990) *Strategic Information Systems, current practices and guidelines*. PhD dissertation. Henley-on-Thames, Henley-The Management College.
- Rezaee, Z., Sharbatoghlie, A., Elam, R. & McMickle, P. L. (2002) Continuous Auditing: Building Automated Auditing Capability. *Auditing*, American Accounting Association, 21 (1), 147.

- Riedy, M. K. & Wen, J. H. (2010) Electronic surveillance of Internet access in the American workplace: implications for management. *Information & Communications Technology Law*, Routledge, 19 (1), 87-99.
- Robertson, J. C. (2000) *Fraud Examination for Managers and Auditors*. Austin, TX, Viesca Books.
- Robinson, N. (1999) The use of focus group methodology -- with selected examples from sexual health research. *Journal of Advanced Nursing*, 29 (4), 905-913.
- Romney, M. B. & Steinbart, P. J. (2009) *Accounting Information Systems*. Eleventh, New Jersey, Uppel Saddle River.
- Rothenberger, M. A., Srite, M. & Jones-Graham, K. (2010) The impact of project team attributes on ERP system implementations: A positivist field investigation. *Information Technology & People*, Emerald Group Publishing Limited, 23 (1), 80-109.
- Rutgers (2010) The Emerging Role of Audit Analytics - Internal Audit Should Embrace Data Analytics *CAR-Lab*, <http://raw.rutgers.edu/node/89>. Accessed: 12/11/2010
- SAP-AG (2009) SAP Library. SAP AG.
- SAP (2010) SAP Named Worldwide Market Share Leader in Business Intelligence, Analytics and Performance Management Software by Top Industry Analyst Firm, *SAP-AG*, <http://www.sap.com/australia/search/index.epx?q1=fraud+detection&num=10>. Accessed: 27/10/2010
- Selby, R. W. (2009) Analytics-Driven Dashboards Enable Leading Indicators for Requirements and Designs of Large-Scale Systems. *Software, IEEE*, 26 (1), 41-49.
- Shuttleworth, M. (2008) Experimental Errors: Type I Error and Type 2 Error, *Experiment-Resources.com*, <http://www.experiment-resources.com/type-I-error.html>. Accessed: 16/06/2011
- Singh, K. H., Best, P. J. & Mula, J. M. M. (2011) Vendor Fraud Detection in Enterprise Systems: An Automated Approach. (*Submitted for publication to International Journal of Auditing*). Blackwell Publishing.
- Singleton, T., Singleton, A., Bologna, J. & Lindquist, R. (2008) *Fraud Auditing and Forensic Accounting*. John Wiley & Sons.
- Singleton, T. W. & Singleton, A. J. (2007) Why don't we detect more fraud? *Journal of Corporate Accounting & Finance*, Wiley Subscription Services, Inc., A Wiley Company, 18 (4), 7-10.
- Smith, M. (2003) *Research Methods in Accounting*. 1st Ed., London, Sage Publications.

- Smith, M. (2011) *Research Methods in Accounting*. 1st Ed., London, Sage Publications.
- Srinidhi, B. (1994) The Influence of Segregation of Duties on Internal Control Judgments. *Journal of Accounting, Auditing & Finance*, Greenwood Publishing, 9 (3), 423-444.
- Stake, R. (1995) *The art of case research*. Thousand Oaks, Sage Publications.
- Standards Australia (2008) Australian Standard AS 8001-2008 - Fraud and Corruption Control, <http://www.saiglobal.com/shop/Script/search.asp>. Accessed: 15/01/2010
- Sutherland, E. H. (1940) *American Sociological Review*, Sage Publications Inc., 5 (1), 1-12.
- Tackett, J. A. (2007) Digital analysis: A better way to detect fraud. *Journal of Corporate Accounting & Finance (Wiley)*, John Wiley & Sons, Inc., 18 (4), 27-36.
- Tatum, M. (2010) What is an Audit Trail, *wiseGEEK*, <http://www.wisegeek.com/what-is-an-audit-trail.htm>. Accessed: 11/11/2010
- TechTarget (2010) Data Visualization, *TechTarget*, <http://searchbusinessanalytics.techtarget.com/definition/data-visualization>. Accessed: 21/12/2011
- UK Treasury (2006) Fraud Report 2005-2006. An analysis of reported fraud in Government Departments., http://www.hm-treasury.gov.uk/media/3/8/fraud_report_government_depts_05-06.pdf Accessed: 10/08/2008
- USDoHHS (1997) General Principles of Software Validation; Final Guidance for Industry and FDA Staff., Center for Devices and Radiological Health.
- Vanasco, R. R. (1998) Fraud auditing *Managerial Auditing Journal* 13 (1), 4-71.
- Vasarhelyi, M. A., Alles, M. G., Kogan, A. & O'Leary, D. (2004) Principles of Analytic Monitoring for Continuous Assurance. *Journal of Emerging Technologies in Accounting*, American Accounting Association, 1, 1-21.
- Vasarhelyi, M. A. & Halper, F. B. (1991) The Continuous Audit of Online Systems. *Auditing*, American Accounting Association, 10 (1), 110-125.
- Velastin, S. A. (1991) An approach to modular programming in C. *Software Engineering for Real Time Systems, 1991., Third International Conference on*.
- Vogel, A. & Kimbell, I. (2005) *ERP for Dummies*. John Wiley & Sons

- Wailgum, T. (2008) Why ERP Systems Are More Important Than Ever, *CXO Media*, http://www.cio.com/article/177300/Why_ERP_Systems_Are_More_Important_Than_Ever?page=2&taxonomyId=3000. Accessed: 09/11/2010
- Wallace, D. R., Ippolito, L. M. & Cuthill, B. (1996) NIST Special Publication 500-234. Reference Information for the Software Verification and Validation Process, <http://hissa.nist.gov/HHRFdata/Artifacts/ITLdoc/234/val-proc.html>. Accessed: 14/11/2011
- WAToday (2009) Card skimming threat lingers: fraud squad, <http://www.watoday.com.au/wa-news/card-skimming-threat-still-lingers-fraud-squad-20091214-krwf.html>. Accessed: 09/01/2012
- Weber, R., A (1999) *Information Systems Control and Audit*. Upper Saddle River, NJ, Prentice Hall.
- Webster (2001) *Webster's New World College Dictionary*. 4th ed., Cleveland, IDG Books Worldwide.
- Wells, J. T. (2002a) Billing schemes, part 1: Shell companies that don't deliver. *Journal of Accountancy*, 194 (1), 76-79.
- Wells, J. T. (2002b) Billing schemes, part 3: Pay-and-return invoicing. *Journal of Accountancy*, 194 (3), 96-98.
- Wells, J. T. (2004) Small Business, Big Losses. *Journal of Accountancy*, 198 (6), 42-47.
- Wells, J. T. (2008) *Principles of Fraud Examination*. 2nd Ed., John Wiley & Sons
- Wells, J. T. (2011) *Principles of Fraud Examination*. 3rd Ed., John Wiley & Sons
- Xu, H., Nord, J. H., Brown, N. & Nord, G. D. (2002) Data quality issues in implementing an ERP. *Industrial Management & Data Systems*, Emerald MCP UP, 102 (1), 47-58.
- Yin, R. K. (2002) *Case Study Research. Design and Methods*. 3rd Edition, California, Sage Publications.
- Yin, R. K. (2009) *Case Study Research. Design and Methods*. 4th Edition, California, Sage Publications.
- Young, M. D. (2011) ELECTRONIC SURVEILLANCE IN AN ERA OF MODERN TECHNOLOGY AND EVOLVING THREATS TO NATIONAL SECURITY. *Stanford Law & Policy Review*, Stanford Law & Policy Review, 22 (1), 11-39.

ooOoo

APPENDICES

Appendix 1: Fraud cases

1. Changes in vendor bank details

A member of staff employed in a finance post exploited access to and knowledge of the supplier database and payment systems to create a new supplier record including a new bank account which appeared to relate to a satellite site for an existing supplier (though it was actually one for a similarly named company registered by the employee) and to raise and pay two illegitimate invoices for circa £81K to the newly registered bank account. In addition it is believed that the staff member inadvertently changed the bank account details of the original supplier and caused two legitimate payments intended for the original supplier to be redirected to the newly registered bank account. These redirected payments totalled £9.2 million. The failure of these to reach the intended recipient caused enquiries to be made which brought the matter to light (UK Treasury 2006).

2. Duplicate invoices

During a routine meeting with numerous employees the newly appointed internal auditor at a dental supply wholesaler discovered that one of the accounting clerks handled invoice processing as well as the occasional overpayment received in the mail. This was clearly a breach of security and needed immediate attention. Further investigation revealed that Veronica, the accounting clerk in question, was processing certain invoices twice. When confronted, she confessed and admitted that her favourite target was her employer's largest supplier, a dental appliance

manufacturer that printed its simple invoices in black ink on plain paper. When she required money she made a copy of the manufacturers invoice before stamping the original. The two were almost indistinguishable. Then she would process the first invoice, send it for approval, and process the invoice again in a few days using the copy she had made. The company would pay the bill twice. When the supplier realised the overpayment, it sent a refund cheque that landed on Veronica's desk, which she took home, got her husband to forge the company's endorsement with a specially made rubber stamp and deposited the cheque into his business account. In less than two years Veronica had embezzled more than US\$250,000 (Wells 2002b).

3. Fake vendor

The IIA San Francisco Chapter (1993) reported a case of non-existent vendor fraud. In this instance there was no verification of existing vendors and cheques were returned to the requesting sales people for delivery to vendors. The audit findings showed that the sales manager had stolen US\$430,000 by means of cheques written to non-existent vendors and diverted to his bank account (Vanasco 1998).

4. Fake invoices

A secretary for a public company interceded on behalf of an unpaid legitimate vendor and the accounts payable department could not locate the original invoice, it nonetheless agreed to pay the vendor based on a fax copy. The secretary seized this opportunity and together with two non-employee accomplices set up three phony companies, and submitted fax copies of doctored original invoices for "consulting fees". The fraud was discovered when a manager questioned a huge variation in the budget – but not until four years and US\$1.7 million later (Wells 2004).

5. Shell Companies (with address same as employee)

A creative writer at a large advertising company opened a business account in the name of SJR Enterprises (a title reflecting his initials), operating from his girlfriends address. He printed an invoice in the name of SJR Enterprises on his home computer. He then colluded with his girlfriend, who worked in the accounting department of the same firm, and following her instructions, billed their employer for US\$4,900 for "services performed under contract 15-822," a description similar to that found on many other invoices. The amount was chosen because the company rarely scrutinised invoices for amounts less than US\$5,000. The girlfriend then created a new vendor file and phony documents to go with it. Once SJR Enterprises was recorded as a vendor, the girlfriend simply put the invoice into a stack of much larger invoices for processing and payment. The scheme worked so well that the pair tried it numerous times successfully. Ultimately the pair defrauded the company of almost US\$700,000 over two years before the scheme was discovered by internal auditing (Wells 2002a).

6. Security firm boss pleads guilty to \$1.4 million fraud

(from: National Business Review, 27 January 2011)

The general manager of an Auckland security equipment supplier has been sentenced to three years and three months in jail after pleading guilty to Serious Fraud Office charges involving a \$1.4 million fraud.

Martyn Tewsley Scott, 51, used his access to National Fire and Security's accounting system to transfer \$1.4 million to his own bank accounts and pay personal invoices amounting to \$6,243.

Seven charges were laid by the SFO in September, relating to accessing a computer for dishonest purpose. Scott worked at the Penrose-based firm for six years.

The SFO said Scott committed the fraud by diverting genuine supplier invoices to his bank account – preventing creditors from being paid, and making double-paying genuine invoices, with the second payment going to one of his bank accounts. He also created false supplier invoices to support other payments covertly paid to his bank account. Scott's early guilty plea and a repayment of almost \$700,000 to National Fire and Security were taken into account in the sentencing today.

Yesterday, an employee from an air conditioning company in Gore plead guilty to six SFO charges relating to the theft of \$600 thousand. John William Jackson, 61, tapped into the computer system of air conditioning company Aire Res-Comm, where he was a director, to divert \$604,779.87 into his own account over a three-year period. (Bond 2011)

Appendix 2: Expert panel protocol

Expert Protocol Validation of Fraud Detection Prototype	
Moderator: _____	Date: _____
No. of Participants: _____	Time: _____
Assistant: _____	Venue: _____
<p>Ask participants to arrive 10-15 minutes early for registration. Room must be set up at least 30 minutes prior (incl. all equipment installation and testing). Each participant will be welcomed by the moderator. Tea and coffee to be served.</p>	
<p>Stage 1: Greeting and Introduction by Moderator - 5 minutes</p> <p>§ Purpose: Welcome participants and express appreciation</p> <p>§ Things to include in welcome:</p> <ul style="list-style-type: none">- Introduce self – first name basis- Introduce the purpose of the group meeting	
<p>Stage 2: Utilities - 2 minutes (combined 7 minutes)</p> <p>§ Purpose: Set the stage for the session</p> <ul style="list-style-type: none">- Confidentiality: highlight definition of confidentiality in the context of the study and information being provided by the group- Recording: highlight presence of audio/video recording equipment <p style="padding-left: 40px;">§ Script: <i>These sessions are being recorded in order to gain the maximum information from the comments you make. The recordings will be used only in strict confidentiality. Your comments will only be used for improvement of the prototype developed for this study.</i></p> <ul style="list-style-type: none">- Observers/assistant: highlight their purpose, introduce them, assure confidentiality	
<p>Stage 3: Icebreaker - 2 minutes (combined 9 minutes)</p> <p>§ Purpose: Preliminary fun question that every one can relate to. Develops rapport, comfort, and an initial relationship.</p>	

- Allow chatter, then quickly refocus without talking over anyone.

Stage 4: Introduce Research Study - 10 minutes (combined 19 minutes)

- § Purpose: introduce the research study
 - PowerPoint presentation
 - § background to the study
 - § work accomplished thus far in the study
 - § demonstrate the prototype– inform participants that they will have an opportunity to individually use the software

Stage 5: Hands-On Session - 20 minutes (combined 39 minutes)

- § Purpose: users get an opportunity to use the software – moderator interacts with individuals during this session

Stage 6: Feedback/Discussion Session 12 minutes (combined 51 minutes)

- § Purpose: users discuss their experience and provide feedback on the software
 - Inform users that they are evaluating software on ; *operation, reporting and visualisations, accuracy & efficiency, and impact on auditor productivity.*

Stage 7: Summary - 5 minutes (combined 56 minutes)

- § Purpose: summarise pertinent points noted in the discussion – ensures no points have been overlooked

Stage 8: Closing thank participants 4 minutes (combined 60 minutes)

- § Purpose: express appreciation
 - Things to include:
 - § Emphasise the importance of their comments
 - § Assure that their ideas will count towards refining the software
 - § Communicate that results will be made available
 - § Dismiss participants with a big Thank You

Stage 9: Wrap up 20-30 minutes

- § Purpose: collect materials
 - Ensure that all materials and recordings are collected
 - Ensure venue is in a tidy state and all equipment returned

Appendix 3: Prototype evaluation questionnaire

EVALUATION OF PROTOTYPE FRAUD DETECTION SOFTWARE

Dear Respondent: Thank you for taking the time to complete this questionnaire. This software is in the prototype stage and your feedback and comments will assist in improving it. Some features may not yet be implemented.

Demographic Information (× all that apply)

Are you a: CPA IT Professional Other _____

Member of: ISACA CISA Other _____

(Please **S** one option only)

#	Item	Rating						
		1	2	3	4	5	6	7
1.	Operation	1	2	3	4	5	6	7
1.1	Easy to use	1	2	3	4	5	6	7
1.2	User-friendly	1	2	3	4	5	6	7
1.3	Navigation of user interface is simple	1	2	3	4	5	6	7
1.4	Onscreen instructions/ help is adequate	1	2	3	4	5	6	7
1.5	Data entry is straightforward	1	2	3	4	5	6	7
2.	Reports							
2.1	Easy to understand	1	2	3	4	5	6	7
2.2	Contains adequate information	1	2	3	4	5	6	7
2.3	Helpful in identifying potential fraud	1	2	3	4	5	6	7
2.4	Are an important tool in a fraud investigators toolkit	1	2	3	4	5	6	7
3.	Visualisations (charts & diagrams)							
3.1	Easy to understand	1	2	3	4	5	6	7
3.2	Useful in aggregating an enormous amount of information	1	2	3	4	5	6	7
3.3	Enables effective exploration of data in a graphical format	1	2	3	4	5	6	7
3.4	Enables identification of relationships or patterns in data that are otherwise difficult to do in textual data	1	2	3	4	5	6	7
3.5	Enhances investigation and analysis for potential fraud	1	2	3	4	5	6	7
3.6	Are an innovative way of presenting information	1	2	3	4	5	6	7
3.7	Are an important tool in a fraud investigators toolkit	1	2	3	4	5	6	7

4. Auditor Productivity
(Assume 30 000 transactions during period of review)

Question 4.1 is based on person days (U= impractical)

4.1 How long would it take to review for fraud, if done...

Manually	<1	1	3	5	10	20+	U
Using other software, e.g. MS Access, MS Excel, ACL, etc.	<1	1	3	5	10	20+	U
Using this software, give an estimate	<1	1	3	5	10	20+	U

4.2 *Based on your response to 4.1, this software may reduce time taken to identify potential fraud in an organisation*

5. Accuracy, Efficiency and Performance

5.1 Produces quality results that are useful in identifying potential fraud

#	Item	Rating						7
		1	2	3	4	5	6	Strongly Agree
5.2	Results are accurate and dependable	1	2	3	4	5	6	7
5.3	Produces the same results as a human expert	1	2	3	4	5	6	7
5.4	Generates results much faster than doing a similar task manually	1	2	3	4	5	6	7
5.5	Is an improvement over basic analysis as it replaces blind querying of data with contextual analysis	1	2	3	4	5	6	7
5.6	Significantly enhances the internal auditing process	1	2	3	4	5	6	7
5.7	Potential to save costs due to improved fraud detection							
5.8	Potential to reduce future fraud by early detection of suspect user activity	1	2	3	4	5	6	7

6. Overall Evaluation

6.1 This software represents substantial advances over other tools currently available in the market

6.2 If available, I am likely to use this software

6.3 If available, I am likely to recommend this software to others

6.4 Overall, this software is a useful auditing tool

7. Comments

7.1 Features of the software you found useful

7.2 Features of the software that could be improved

7.3 Suggest any additional features to include in the software

7.4 Do you currently use a software tool for auditing?

Yes No

If Yes, please provide details below

7.5 Other comments

7.6 Optional: Should you wish to receive more information on this software or research please complete contact information

Name: _____

Email: _____

Thank you for completing the questionnaire
© K Singh

Appendix 4: Prototype menu navigation



Figure A4.1: Start-up screen

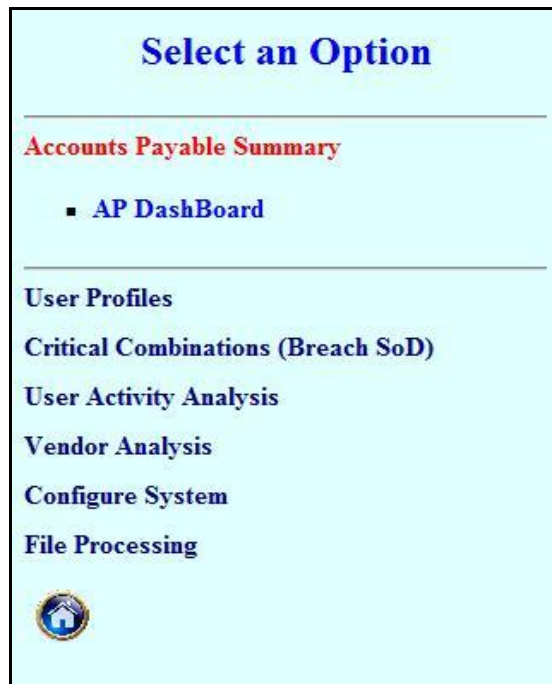


Figure A4. 2: Accounts payable summary menu



Figure A4. 3: User profiles menu



Figure A4. 4: Critical combinations menu



Figure A4. 5: User activity analysis menu

Detailed User Activity Analysis

INSTRUCTIONS

1. Enter a User
2. Click Run SAS Program
4. Wait for SAS to Finish
5. Click View Report

Enter User:

[View Reports](#)

[Run SAS Program](#)

[Run Visualization](#)

Figure A4. 6: Detailed user activity analysis menu

User Activity Analysis REPORTS

INSTRUCTIONS

1. Click a report link below.
2. Click **Back** in the report screen to return here.

Transaction Reports for this User	Visualizations for this User
<ol style="list-style-type: none">1. Bank Account Changes2. Summary of Invoice Transactions3. Summary of Payment Transactions4. Details of Invoice Transactions5. Details of Payment Transactions6. Round Dollar Invoices7. Round Dollar Payments8. Invoices 5% Below Approval Limit9. Payments 5% Below Approval Limit10. Duplicate Transactions	<ol style="list-style-type: none">1. Vendors Touched by this User2. Analyse interaction between this User and selected Vendor <p>Enter Vendor Id: <input type="text"/></p> <p><input type="button" value="Run SAS Program"/> <input type="button" value="View Report"/></p>

Figure A4. 7: User activity reports menu

Select an Option

Accounts Payable Summary

User Profiles

Critical Combinations (Breach SoD)

User Activity Analysis

Vendor Analysis

- **Analyse Vendor Transactions**

- **Benfords Law: Vendor Invoices**
- **Benfords Law: Vendor Payments**
- **Benfords Law: Analysis**

- **Vendors: Multiple Bank Changes**
- **Vendors: All Bank Changes**

- **Vendors: Sharing Bank Accounts**
- **Vendors: Multiple Bank Accounts**
- **Vendors: Multiple Master Records**

- **TOP 5 Vendors: Sum of Invoices**
- **TOP 5 Vendors: Sum of Payments**
- **All Vendors: Sum of Invoices**
- **All Vendors: Sum of Payments**

- **Vendors: Similar Names**
- **Complete Vendor List**

Figure A4. 8: Vendor analysis menu

Analyse Vendor Transactions

INSTRUCTIONS

1. Enter a Vendor Id
2. Click Run SAS Program & wait for SAS to Finish
4. Click Run Visualization & wait for SAS to Finish
5. Click View Reports

Enter Vendor Id:

[View Reports](#)

[Run SAS Program](#)

[Run Visualization](#)

Figure A4. 9: Analyse vendor transactions menu

Analysis of Vendor Transactions

INSTRUCTIONS

1. Click a report link below.
2. Click **Back** in the report screen to return here.

Transaction Reports for this Vendor	Visualizations for this Vendor
<ol style="list-style-type: none">1. Summary Transaction Statistics2. Bank Account Changes3. Transaction History4. Duplicate Transactions	<ol style="list-style-type: none">1. Analysis of Vendor Payments <i>(to identify Payments that vary by large amounts)</i>

Figure A4. 10: Analysis of vendor transactions (reports) menu

Search Vendor for Similar Names

INSTRUCTIONS

1. Enter a **Vendor Name**
2. Click **Run SAS Program**
3. Wait for SAS to **Finish**
4. Wait for SAS to **Finish**
5. Click **View Report**

Search Tips

- i) Use % for **wild-card** search

TEL% to search for all names beginning with TEL

%TEL for all names ending in TEL

%TEL% for all names that contain the substring TEL

- ii) Use _ for a **single wild-card** character

HARVEY_W% for all names like HARVEY W

Enter Vendor Name:

Run SAS Program

View Report

Figure A4. 11: Search vendor menu

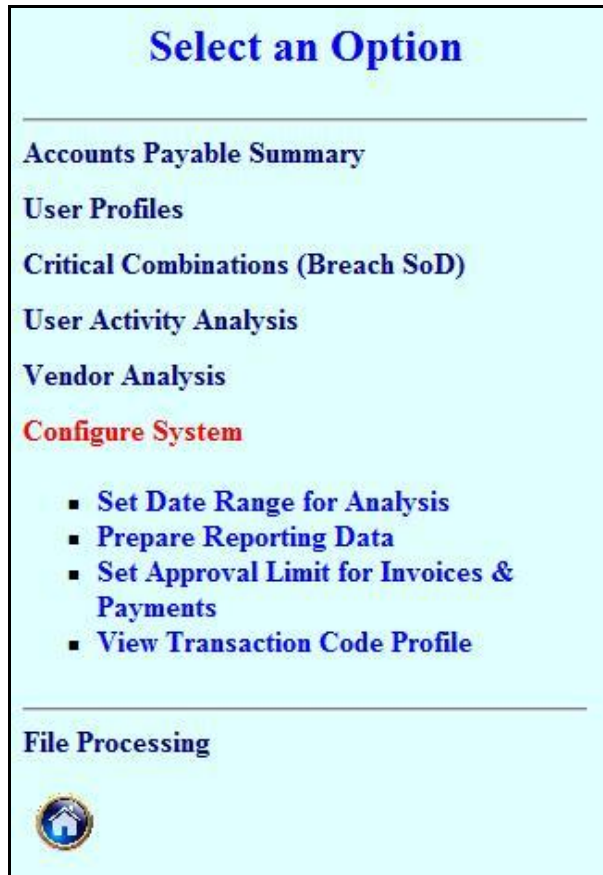


Figure A4. 12: Configure system menu

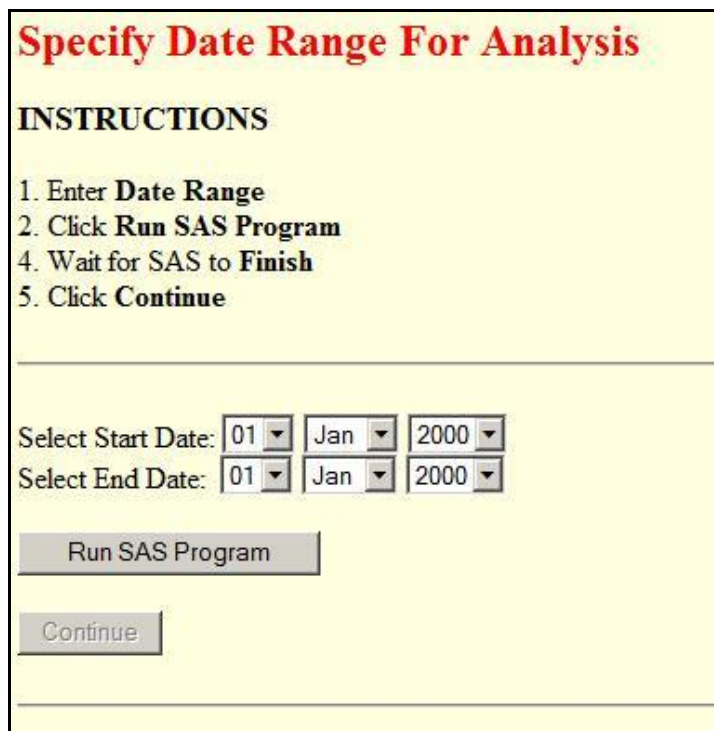


Figure A4. 13: Set date range for analysis menu

Set Approval Limit for Invoices & Payments

INSTRUCTIONS

1. Enter **Amount**
2. Click **Set Limit** when done

Enter Amount:

For example: 1000.00
Do not enter the \$ symbol

Figure A4. 14: Set approval limit for invoices & payments menu

Select an Option

Accounts Payable Summary

User Profiles

Critical Combinations (Breach SoD)

User Activity Analysis

Vendor Analysis

Configure System

File Processing

- Data Conversion & Import



Figure A4. 15: File processing menu

Data Conversion & Import

INSTRUCTIONS

1. Perform **Data Extraction Procedure**
2. **Copy** extracted files to folder: C:\SASDATA\SAP\SAPEXTRACT
3. Click **Run SAS Program**
4. Wait for SAS to **Finish**
5. Click **Continue**

Enter **U** - to Update existing Data Warehouse or
Enter **C** - to Create a Data Warehouse. **This will erase all existing data.**

Run SAS Program

Continue

Figure A4. 16: Data conversion & import menu

Data Conversion & Import

INSTRUCTIONS

1. Perform **Data Extraction Procedure**
2. **Copy** extracted files to folder: C:\SASDATA\SAP\SAPEXTRACT
3. Click **Run SAS Program**
4. Wait for SAS to **Finish**
5. Click **Continue**

Enter **U** - to Update existing
Enter **C** - to Create a Data Warehouse

Run SAS Program

Continue

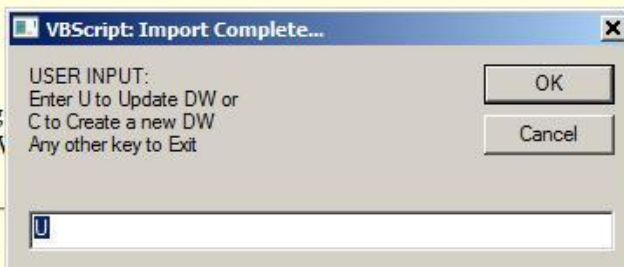


Figure A4. 17: Update/create data warehouse selection screen

Appendix 5: Results from test data

Period of analysis: 01/12/2003 to 31/12 2003 (1 Month)

Analysis procedures

User profiles	Users are profiled to determine the scope of activities they have performed. Activities include vendor maintenance, invoicing and payment transactions. Summary and detailed reports are produced.
Critical combinations	Users that violate segregation of duties are identified and a report of potentially risky users is produced.
User activity analysis	An individual user is identified from the risky users list and selected for detailed investigation. Reports documenting individual user activities are produced.
Vendor analysis	A series of investigations are performed on active vendors, including vendors sharing bank accounts, vendors with multiple bank accounts, vendors with multiple master records, and Benford's law.

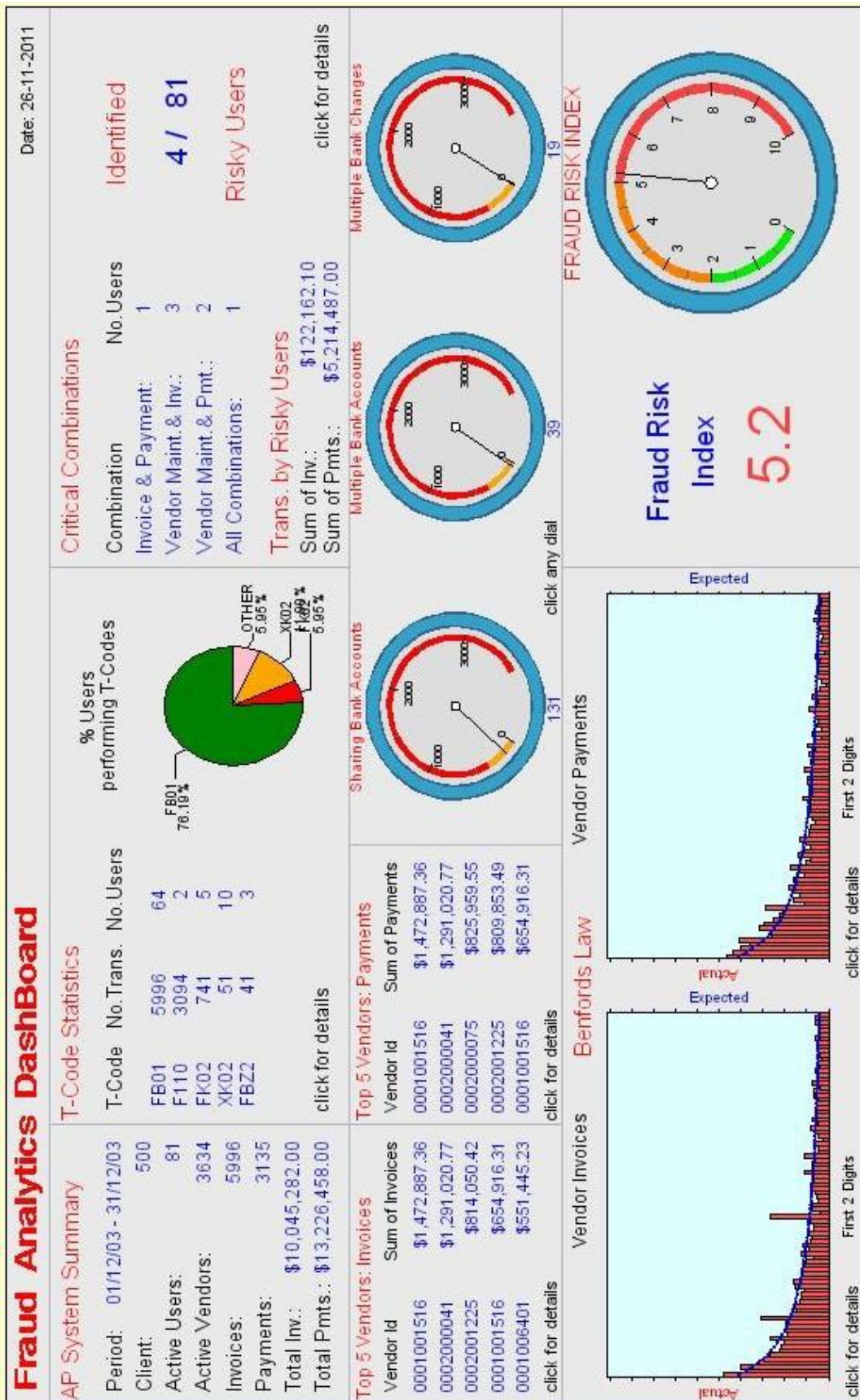


Figure A5.1: Dashboard

User profiling

User Activities Summary		
For Period: 01Dec03 to 31Dec03		
Number of Records Found = 5		
TCODE	Transaction Name	Activity
FB01	Post Document	5996
F110	Parameters for Automatic Payment	3094
FK02	Change Vendor (Accounting)	741
XK02	Change vendor (centrally)	51
FBZ2	Post Outgoing Payments	41
N = 5		

Figure A5. 2: User activities summary

User Profile: Vendor Maintenance		
For Period: 01Dec03 to 31Dec03		
Number of Records Found = 15		
User	Transaction Code	
	FK02	XK02
1USRARSCP	3	16
1USRLEML	3	5
1USRAGNKJ	628	.
1USRATKSE	106	.
1USRINGGP	.	12
1USRRVIMJ	.	7
1USRICKSZB	.	3
1USRMARDE	.	3

Figure A5. 3: User profile – vendor maintenance

User Profile: Invoice Transactions
 For Period: 01Dec03 to 31Dec03
 Number of Records Found = 64

User	Transaction Code	
	FB01	
1USRILLSJ		40
1USRLENLZ		465
1USRARMKG		318
1USRACOMJ		308
1USROLAMS		351
1USRYNAAE		214
1USROGERT		328
1USRULLME		188

Figure A5. 4: User profile – invoice transactions

User Profile: Payment Transactions
 For Period: 01Dec03 to 31Dec03
 Number of Records Found = 5

User	Transaction Code	
	FBZ2	F110
1USRCCRBD	.	1994
1USRATTNC	.	1100
1USRARSCP	38	.
1USRLLEML	2	.
1USRPEIBA	1	.
TOTAL	41	3094

Figure A5. 5: User profile – payment transactions

User Profile: Invoices or Payment Transactions			
For Period: 01Dec03 to 31Dec03			
Number of Records Found = 69			
User	Transaction Code		
	FBZ2	FB01	F110
1USRARSCP	38	7	.
1USRILLSJ	.	40	.
1USRCCRBD	.	.	1994
1USRATTNC	.	.	1100
1USRLENLZ	.	465	.
1USRARMKG	.	318	.
1USRACOMJ	.	308	.
1USROLAMS	.	361	.

Figure A5. 6: User profile – invoices or payment transactions

User Profile: All Combinations*					
For Period: 01Dec03 to 31Dec03					
Number of Records Found = 84					
User	Transaction Code				
	XK02	FK02	FBZ2	FB01	F110
1USRARSCP	16	3	38	7	.
1USRILLSJ	.	.	.	40	.
1USRLEML	5	3	2	.	.
1USRREVPW	1	.	.	16	.
1USRCCRBD	1994
1USRATTNC	1100
1USRLENLZ	.	.	.	465	.
1USRARMKG	.	.	.	318	.

Figure A5. 7: User profile – all combinations

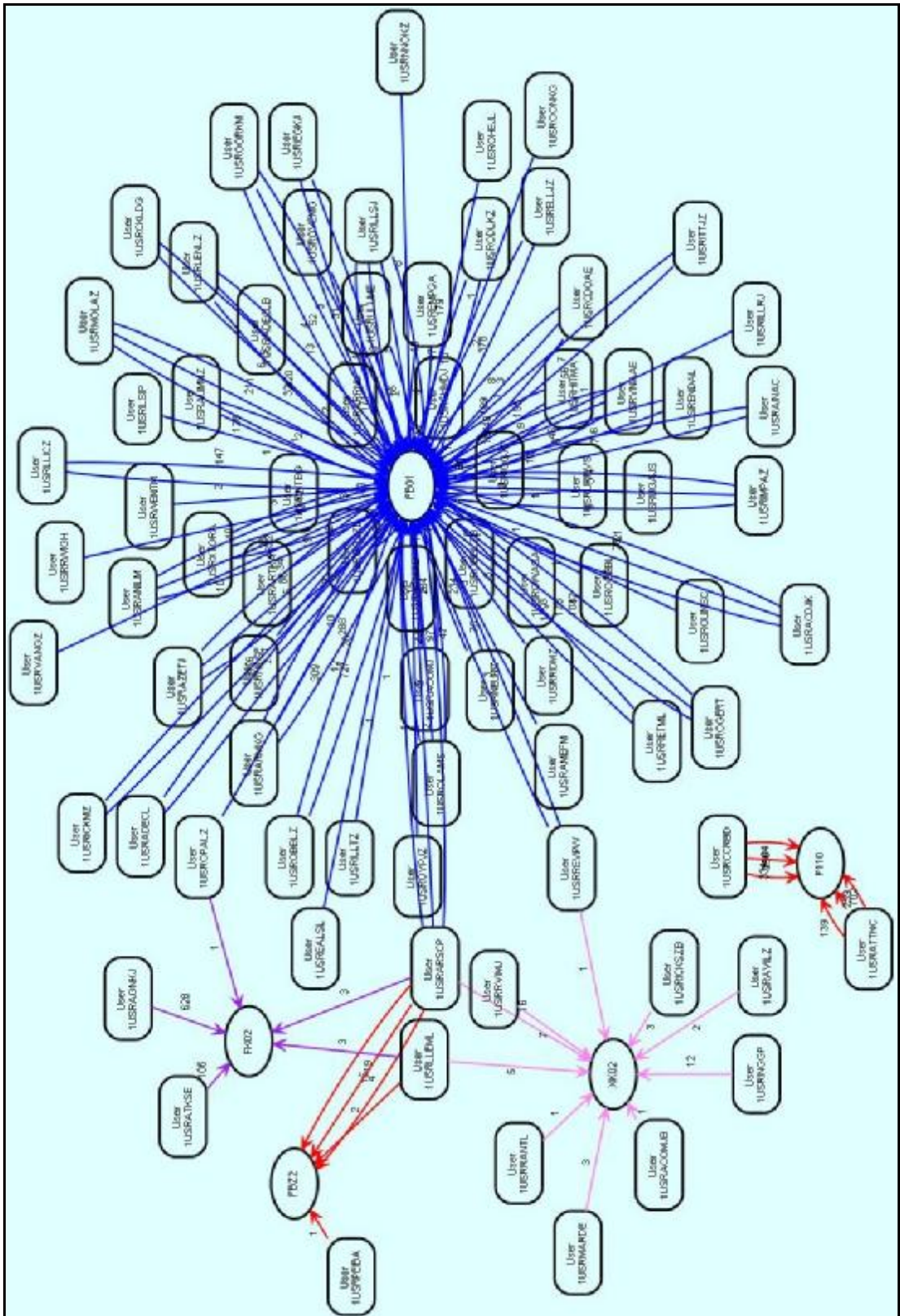


Figure A5. 8: Visualisation – all combinations

Critical combinations

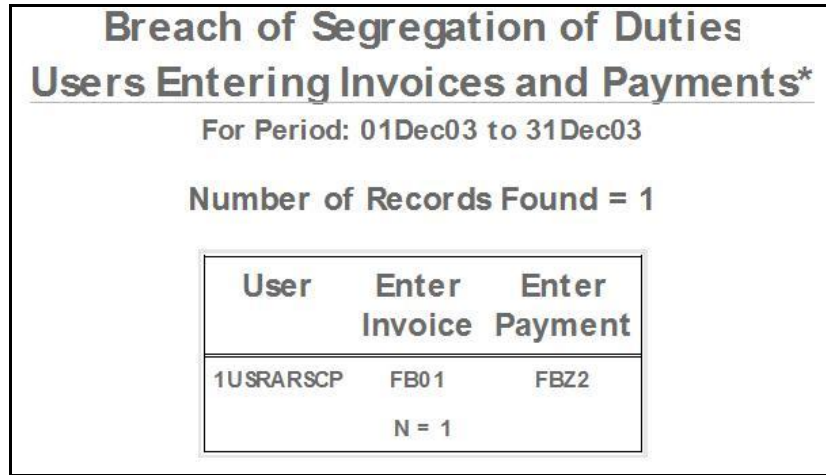


Figure A5. 9: Violation of SoDs – users entering invoices and payments

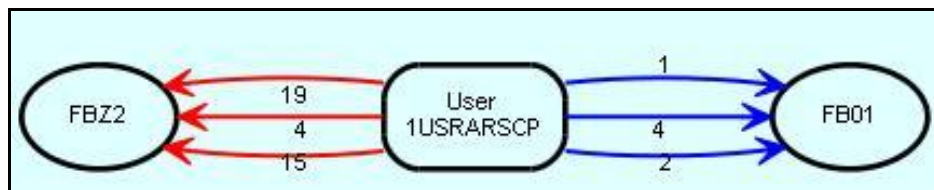


Figure A5. 10: Visualisation - users entering invoices and payments

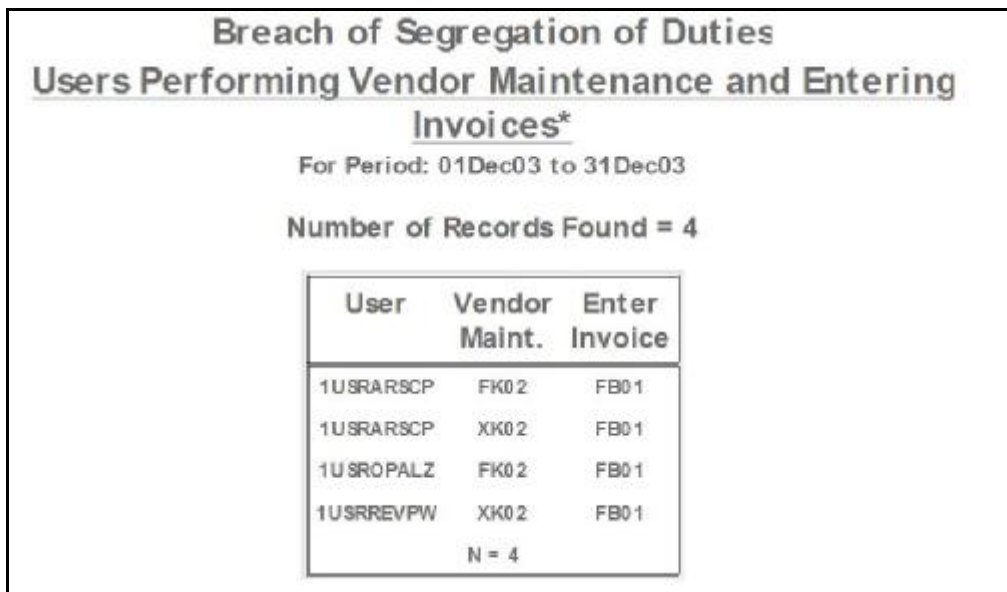


Figure A5. 11: Violation of SoDs – users performing vendor maintenance and entering invoices

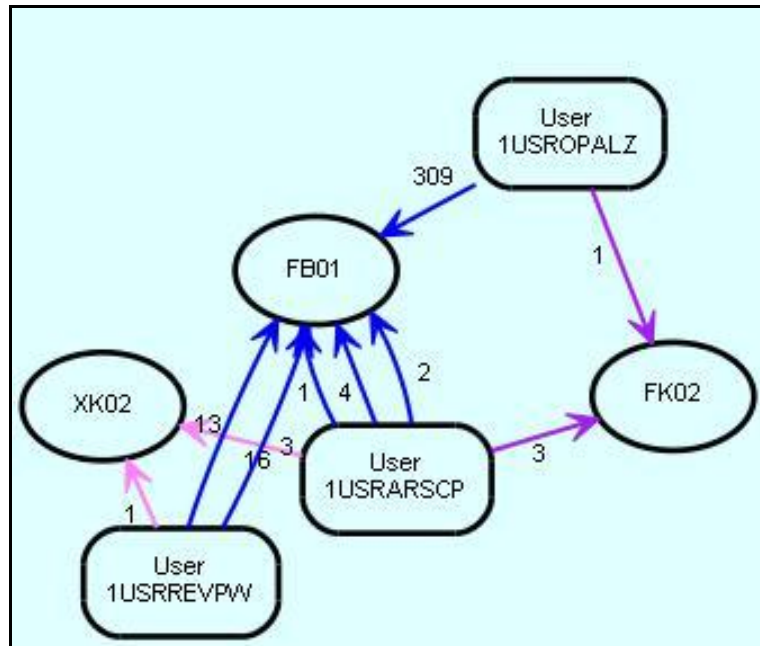


Figure A5. 12: Visualisation - users performing vendor maintenance and entering invoices

Breach of Segregation of Duties
Users Performing Vendor Maintenance and Entering
Payments*

For Period: 01Dec03 to 31Dec03

Number of Records Found = 4

User	Vendor Maint.	Enter Payment
1USRARSCP	FK02	FBZ2
1USRARSCP	XK02	FBZ2
1USRLEML	FK02	FBZ2
1USRLEML	XK02	FBZ2
N = 4		

Figure A5. 13: Violation of SoDs – users performing vendor maintenance and entering payments

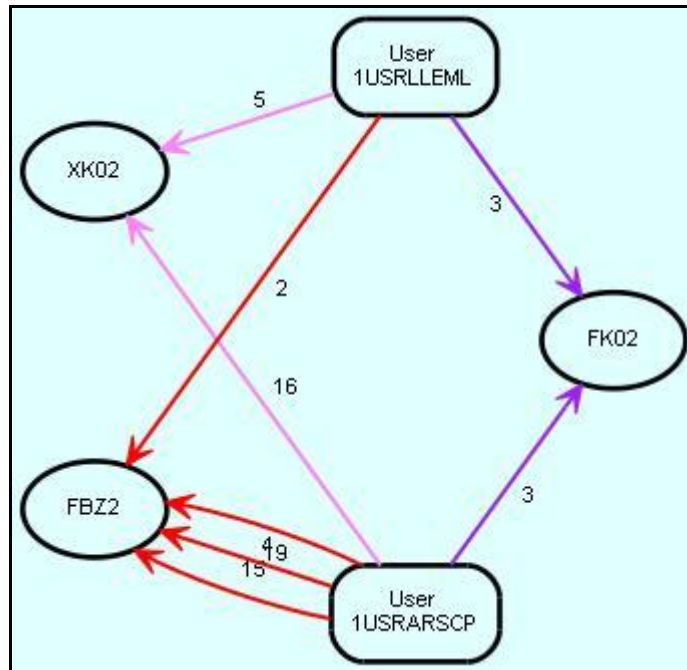


Figure A5. 14: Visualisation - users performing vendor maintenance and entering payments

Breach of Segregation of Duties
Users Performing Vendor Maintenance, Entering Invoices and Payments*
 For Period: 01Dec03 to 31Dec03

Number of Records Found = 2

User	Vendor Maint.	Enter Invoice	Enter Payment
1USRARSCP	FK02	FB01	FBZ2
1USRARSCP	XK02	FB01	FBZ2

N = 2

Figure A5. 15: Violation of SoDs – users performing vendor maintenance, entering invoices and payments

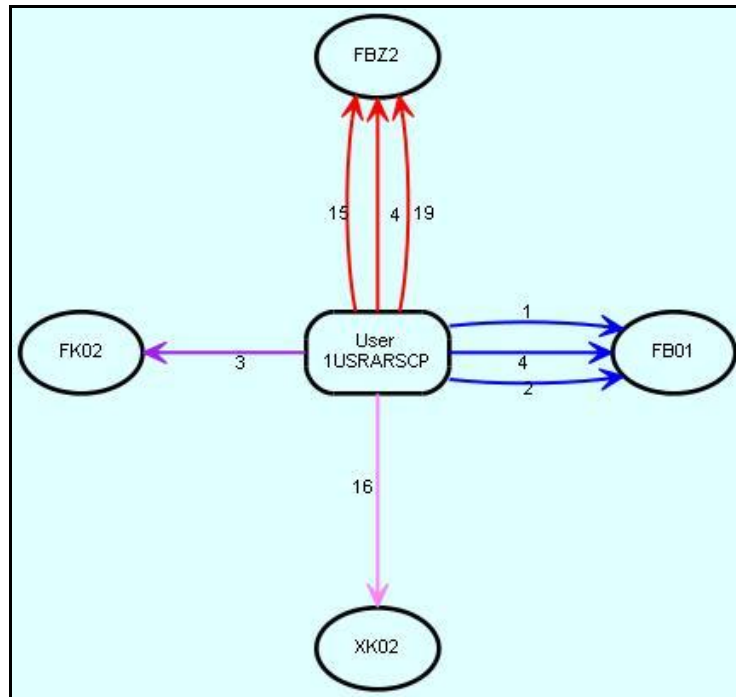


Figure A5. 16: Visualisation - users performing vendor maintenance, entering invoices and payments

User activity analysis

Investigation of user: 1USRARSCP

Bank Account Changes by User: 1USRARSCP
 For Period: 01Dec03 to 31Dec03
 Number of Records Found = 19
[Back](#)

Date	Time	Change No.	Vendor Id	Vendor	Bank Details	TCode
22/12/03	1:28:35	0001995079	0000007200	2VENDOR HARRIS	114-879 066600156	XK02
22/12/03	1:47:09	0001995093	0000001978	2VENDORMERS	734-106 573299	XK02
22/12/03	1:52:29	0001995110	0001008831	2VENDOR GILLET ELECTRICAL	084-484 484049846	XK02
22/12/03	2:16:12	0001995279	0001002028	2VENDORH PTY LTD	014-651 198609187	XK02
22/12/03	2:19:10	0001995377	0001001485	2VENDOR DEVELOPMENTS INTERNATIONAL	033-157 227933	XK02
22/12/03	2:23:25	0001995471	0001028299	2VENDOR DRY	034-187	XK02

Figure A5. 17: Bank account changes by user - 1USRARSCP

Invoice Transactions by User: 1USRARSCP
For Period: 01Dec03 to 31Dec03
Number of Records Found = 7

[Back](#)

Doc.Date	Post.Date	Doc.No.	Co.Code	Vendor Id	Vendor	Amount	TCode
09/12/03	09/12/03	1700001024	ESXS	0000012967	2VENDORINT	\$209.35	FB01
16/12/03	16/12/03	1700001028	ESXS	0000017106	2VENDOR COOK	\$124.80	FB01
16/12/03	16/12/03	1700005337	QASA	0000014414	2VENDORMCMATH	\$247.50	FB01
16/12/03	16/12/03	1700005338	QASA	0001016292	2VENDORLECTRICAL	\$0.33	FB01
18/12/03	18/12/03	1700001410	ESDA	0001018319	2VENDORENDER	\$1.90	FB01
18/12/03	18/12/03	1700005343	QASA	0000013613	2VENDORABRIS	\$227.50	FB01
22/12/03	22/12/03	1700005349	QASA	0000007200	2VENDOR HARRIS	\$551.75	FB01
						\$1,363.13	

N = 7

Figure A5. 18: Invoice transactions by user - 1USRARSCP

Payment Transactions by User: 1USRARSCP
For Period: 01Dec03 to 31Dec03
Number of Records Found = 38

[Back](#)

Doc.Date	Post. Date	Doc.No.	Co.Code	Vendor Id	Vendor	Amount	TCode
01/12/03	09/12/03	0500059070	QASA	0001001470	2VENDORMISSONER OF TAXATION	\$50.20	FBZ2
03/12/03	09/12/03	0500059009	QASA	0001005401	2VENDORIONER OF PAYROLL TAX	\$551,445.23	FBZ2
04/12/03	09/12/03	0000022783	ESDA	0001005401	2VENDORIONER OF PAYROLL TAX	\$46,683.91	FBZ2
04/12/03	09/12/03	0000027893	ESXS	0001005401	2VENDORIONER OF PAYROLL TAX	\$146,662.89	FBZ2
09/12/03	10/12/03	0500027891	ESXS	0001001510	2VENDORIAN TAXATION OFFICE	\$649,713.80	FBZ2
09/12/03	10/12/03	0500027992	ESXS	0001001510	2VENDORUPPORT AGENCY	\$140.00	FBZ2
09/12/03	10/12/03	0500027993	ESXS	0002000041	2VENDOR ACCUMULATED FUND	\$156,071.00	FBZ2
09/12/03	10/12/03	0500027994	ESXS	0002000041	2VENDOR ACCUMULATED FUND	\$76,232.00	FBZ2
09/12/03	10/12/03	0500027995	ESXS	0002000041	2VENDOR ACCUMULATED FUND	\$205.02	FBZ2
09/12/03	10/12/03	0000027996	ESXS	0002000517	2VENDORENT SUPERANNUATION OFFICE	\$130,953.12	FBZ2
09/12/03	10/12/03	0000027997	ESXS	0002000517	2VENDORENT SUPERANNUATION OFFICE	\$509.10	FBZ2
09/12/03	10/12/03	0500027998	ESXS	0001027535	2VENDORATION SERVICES PTY LTD	\$0,532.21	FBZ2
10/12/03	07/01/04	0500090544	QASA	0001027535	2VENDORATION SERVICES PTY LTD	\$110,760.05	FBZ2
10/12/03	07/01/04	0500090545	QASA	0002000042	2VENDOR	\$9,245.12	FBZ2
10/12/03	07/01/04	0500090540	QASA	0002000042	2VENDOR	\$7,865.33	FBZ2
10/12/03	07/01/04	0000090547	QASA	0001001510	2VENDORUPPORT AGENCY	\$20,019.48	FBZ2

Figure A5. 19: Payment transactions by user - 1USRARSCP

Round Dollar Payments to Vendors
 Entered by User: 1USRARSCP
 For Period: 01Dec03 to 31Dec03

Number of Records Found = 4

[Back](#)

Doc. Date	Post. Date	Doc.No.	Co.Code	Vendor Id	Vendor	Amount	Tcode
22DEC2003	06JAN2004	0500023136	ESDA	0001001943	2VENDOROLUNTEER COAST GUARD	\$66.00	FBZ2
22DEC2003	06JAN2004	0500023135	ESDA	0001000835	2VENDORER MARINE RESCUE	\$88.00	FBZ2
22DEC2003	06JAN2004	0500023134	ESDA	0001016629	2VENDORFE SAVING QUEENSLAND	\$550.00	FBZ2
22DEC2003	22DEC2003	1500007433	ESXS	0001008050	2VENDORE CITY COUNCIL	\$6.00	FBZ2

N = 4

Figure A5. 20: Round dollar payments by user - 1USRARSCP

Vendors Touched by User: 1USRARSCP *
 For Period: 01Dec03 to 31Dec03

Number of Records Found = 40

[Back](#)

Vendor Id	TCode			
	XK02	FK02	FBZ2	FB01
000007200	1	.	.	1
000200041	.	.	8	.
0001001516	.	.	4	.
0001001518	.	.	4	.
0001027635	.	.	4	.
000200042	.	.	4	.
0002000517	.	.	4	.
0001006401	.	.	3	.
0000001978	1	.	.	.
0000012967	.	.	.	1
0000013613	.	.	.	1
0000014414	.	.	.	1

Figure A5. 21: Vendors touched by user - 1USRARSCP

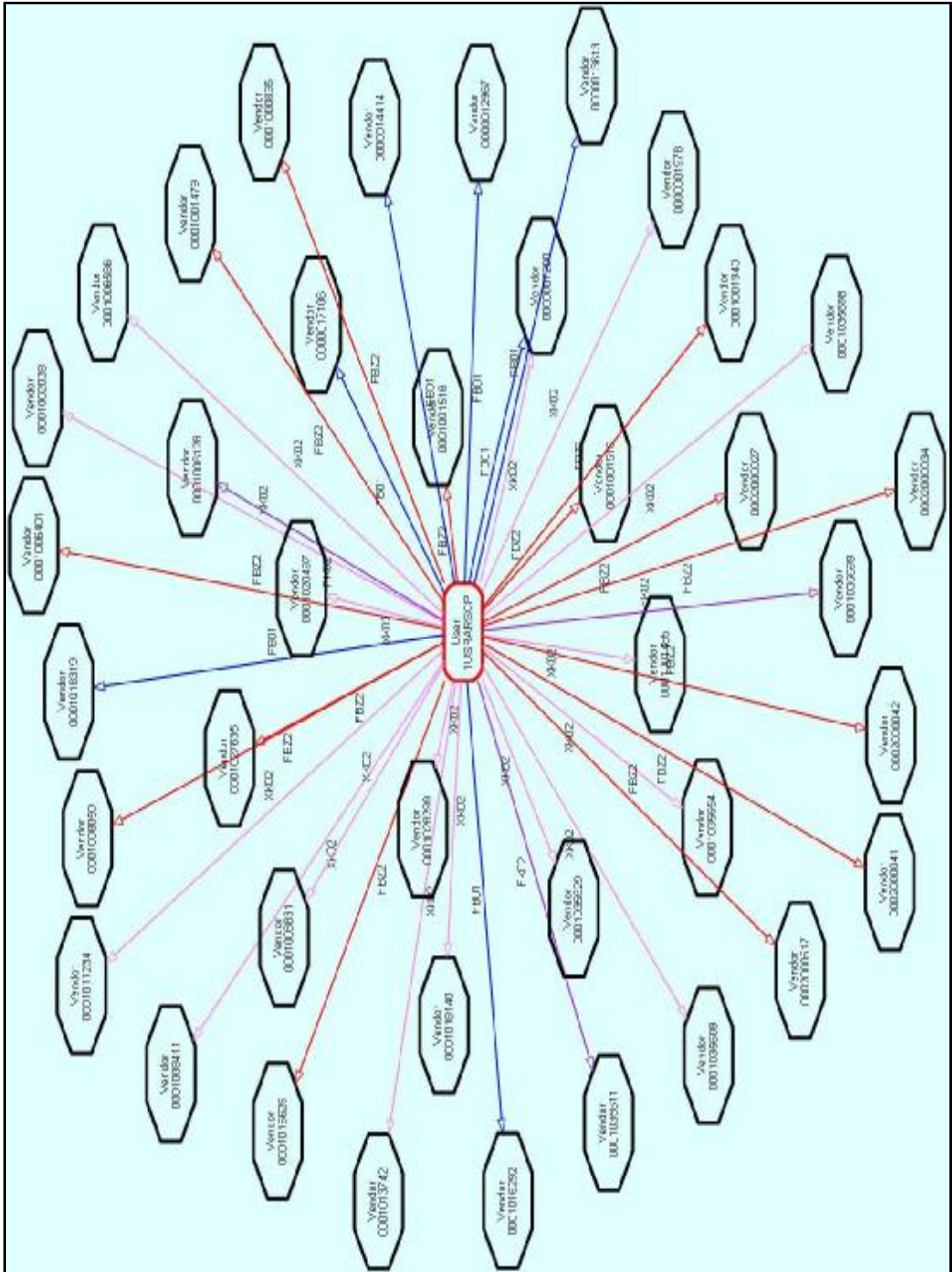


Figure A5. 22: Visualisation – vendors touched by user - 1USRARSCP

User: 1USRARSCP Interacting with Vendor: 0002000041

Transactions

For Period: 01Dec03 to 31Dec03

Number of Records Found = 8

[Back](#)

TCode=FBZ2

Date	Doc. No.	Co.Code	Vendor Id	Amount	User
09/12/03	0500027993	ESXS	0002000041	\$156,071.88	1U SRARSCP
09/12/03	0500027994	ESXS	0002000041	\$70,232.89	1U SRARSCP
09/12/03	0500027995	ESXS	0002000041	\$285.82	1U SRARSCP
10/12/03	0500090549	QASA	0002000041	\$648,337.99	1U SRARSCP
22/12/03	0500028256	ESXS	0002000041	\$165,593.12	1U SRARSCP
22/12/03	0500028257	ESXS	0002000041	\$70,071.25	1U SRARSCP
22/12/03	0500028258	ESXS	0002000041	\$285.82	1U SRARSCP
22/12/03	0500090555	QASA	0002000041	\$642,682.78	1U SRARSCP

N = 8

Figure A5. 23: User 1USRARSCP interacting with vendor 0002000041

Vendor analysis

Vendors Sharing Bank Accounts*									
For Period: 01Dec03 to 31Dec03									
Number of Records Found = 131									
Bank Details=014-504 536293649									
<table border="1"><thead><tr><th>Vendor Id</th><th>Vendor</th></tr></thead><tbody><tr><td>0000004705</td><td>2VENDORLANG</td></tr><tr><td>0000004772</td><td>2VENDORLANG</td></tr><tr><td colspan="2">N = 2</td></tr></tbody></table>		Vendor Id	Vendor	0000004705	2VENDORLANG	0000004772	2VENDORLANG	N = 2	
Vendor Id	Vendor								
0000004705	2VENDORLANG								
0000004772	2VENDORLANG								
N = 2									
Bank Details=014-707 353869879									
<table border="1"><thead><tr><th>Vendor Id</th><th>Vendor</th></tr></thead><tbody><tr><td>0000004774</td><td>2VENDORPIR</td></tr><tr><td>0000015080</td><td>2VENDOR ROWATT</td></tr><tr><td colspan="2">N = 2</td></tr></tbody></table>		Vendor Id	Vendor	0000004774	2VENDORPIR	0000015080	2VENDOR ROWATT	N = 2	
Vendor Id	Vendor								
0000004774	2VENDORPIR								
0000015080	2VENDOR ROWATT								
N = 2									
Bank Details=032-854 32373134									
<table border="1"><thead><tr><th>Vendor Id</th><th>Vendor</th></tr></thead><tbody><tr><td>0000005810</td><td>2VENDORERGU SON</td></tr></tbody></table>		Vendor Id	Vendor	0000005810	2VENDORERGU SON				
Vendor Id	Vendor								
0000005810	2VENDORERGU SON								

Figure A5. 24: Vendors sharing bank accounts

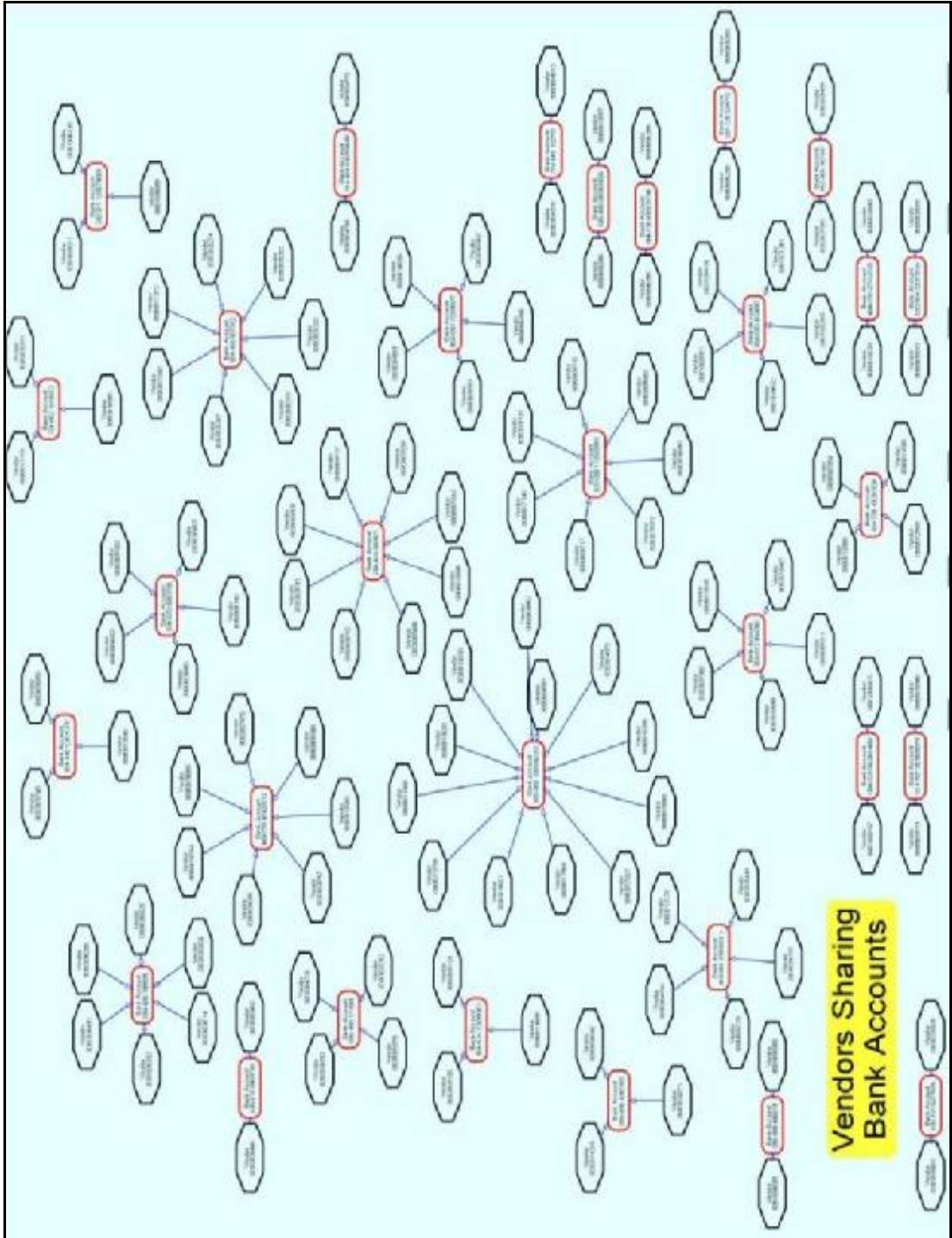


Figure A5. 25: Visualisation - vendors sharing bank accounts

Vendors with Multiple Bank Accounts*

For Period: 01Dec03 to 31Dec03

Number of Records Found = 39

Vendor Id=0000000753

Vendor Name	Bank Details	Date	Time	User
2VENDORLLOYD BUCKLEY	638-050 302582	11/12/03	8:21:03	1USRATKSE
2VENDORLLOYDBUCKLEY	638-0503032582	16/12/03	11:36:52	1USRINGGP

N = 2

Vendor Id=0000004737

Vendor Name	Bank Details	Date	Time	User
2VENDOR WRIGHT	064-81510234130	02/12/03	1:24:47	1USRATKSE
2VENDOR WRIGHT	064-815909571	24/12/03	1:04:21	1USRAGNKJ

N = 2

Vendor Id=0000005045

Vendor Name	Bank Details	Date	Time	User
2VENDORJEFFREY	656-4004141908	24/12/03	9:41:53	1USRAGNKJ

Figure A5. 26: Vendors with multiple bank accounts

Vendors: Multiple Bank Changes

For Period: 01Dec03 to 31Dec03

Number of Records Found = 19

Vendor Id	Vendor Name	No of Changes
0000009646	2VENDORMILE S	3
0000000753	2VENDORLLOYD BUCKLEY	2
0000004737	2VENDOR WRIGHT	2
0000005045	2VENDOR JEFFREY	2
0000005354	2VENDOR MCLEOD	2
0000005610	2VENDOROZA	2
0000006046	2VENDORUGLAS HICKSON	2
0000006286	2VENDORIMMINS	2
0000006299	2VENDORKITSON	2
0000008928	2VENDORJ MULLER	2
0000009412	2VENDORCHARD MUDRA	2
0000009683	2VENDOR SH	2
0000011334	2VENDOROOD	2
0000013624	2VENDOR FERGU SON	2
0000014755	2VENDOR CROSSMAN	2

Figure A5. 28: Vendors with multiple changes to their bank accounts

Vendors: Multiple Master Records

Number of Records Found = 6746

Vendor	No. of Master Records
2VENDOR	713
2VENDOR PTY LTD	160
2VENDORPTY LTD	140
2VENDORAND AMBULANCE SERVICE	138
2VENDORIA POST	127
2VENDORS	100
2VENDOR RURAL FIRE BRIGADE	90
2VENDORE	71
2VENDORRURAL FIRE BRIGADE	63
2VENDOR ELECTRICAL	59
2VENDORATED PEST CONTROL	57
2VENDORTY LTD	54
2VENDORAND FIRE SERVICE	52
2VENDORR	51
2VENDORN	48
2VENDOR SMITH	45

Figure A5. 29: Vendors with multiple master records

TOP 5 Vendors: Sum of Invoices
For Period: 01Dec03 to 31Dec03
Number of Records Found = 5

Co Code	Vendor Id	Vendor	Sum of Invoices	No of Invoices
QA SA	0001001516	2VENDORIAN TAXATION OFFICE	\$1,472,887.36	2
QA SA	0002000041	2VENDOR ACCUMULATED FUND	\$1,291,020.77	2
ESXS	0002001225	2VENDORONE	\$814,050.42	2
ESXS	0001001516	2VENDORIAN TAXATION OFFICE	\$654,916.31	2
QA SA	0001006401	2VENDORIONER OF PAYROLL TAX	\$551,445.23	1
			\$4,784,320.09	
N = 5				

Figure A5. 30: Top 5 vendors by sum of invoices

TOP 5 Vendors: Sum of Payments
For Period: 01Dec03 to 31Dec03
Number of Records Found = 5

Co Code	Vendor Id	Vendor	Sum of Payments	No of Payments
QA SA	0001001516	2VENDORIAN TAXATION OFFICE	\$1,472,887.36	2
QA SA	0002000041	2VENDOR ACCUMULATED FUND	\$1,291,020.77	2
QA SA	0002000075	2VENDOR	\$825,959.55	7
ESXS	0002001225	2VENDORONE	\$809,853.49	3
ESXS	0001001516	2VENDORIAN TAXATION OFFICE	\$654,916.31	2
			\$5,054,637.48	
N = 5				

Figure A5. 31: Top 5 vendors by sum of payments

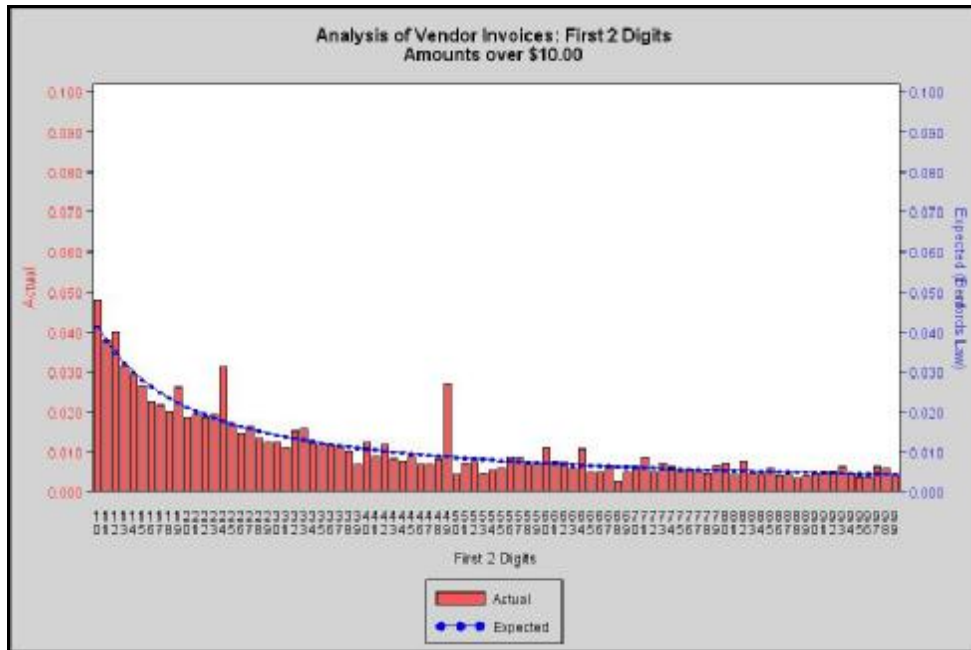


Figure A5. 32: Benford's Law – analysis of vendor invoices

Benfords Law - Analysis of Vendor Invoices
First 2 Digits = 49
 For Period: 01Dec03 to 31Dec03
 Number of Records Found = 116

[Back](#)

User=1USRACDJK

Vendor Id	Doc. No.	Doc. Date	Post. Date	Amount	TCode
0001001796	1900318075	07DEC2003	06JAN2004	\$49.50	FB01
0001002963	1900054816	07DEC2003	08JAN2004	\$490.55	FB01

N = 2

User=1USRACOMJ

Vendor Id	Doc. No.	Doc. Date	Post. Date	Amount	TCode
0001003806	1900317524	02DEC2003	05JAN2004	\$49.15	FB01
0001013418	1900321883	04DEC2003	30JAN2004	\$49.50	FB01
0001006339	1900317923	10DEC2003	06JAN2004	\$49.50	FB01
0001003806	1900322157	11DEC2003	02FEB2004	\$49.57	FB01
0001000010	1900317302	18DEC2003	23DEC2003	\$49.50	FB01
0001003806	1900322160	28DEC2003	02FEB2004	\$49.57	FB01

Figure A5. 33: Benford's Law – investigation of spike at digit 49

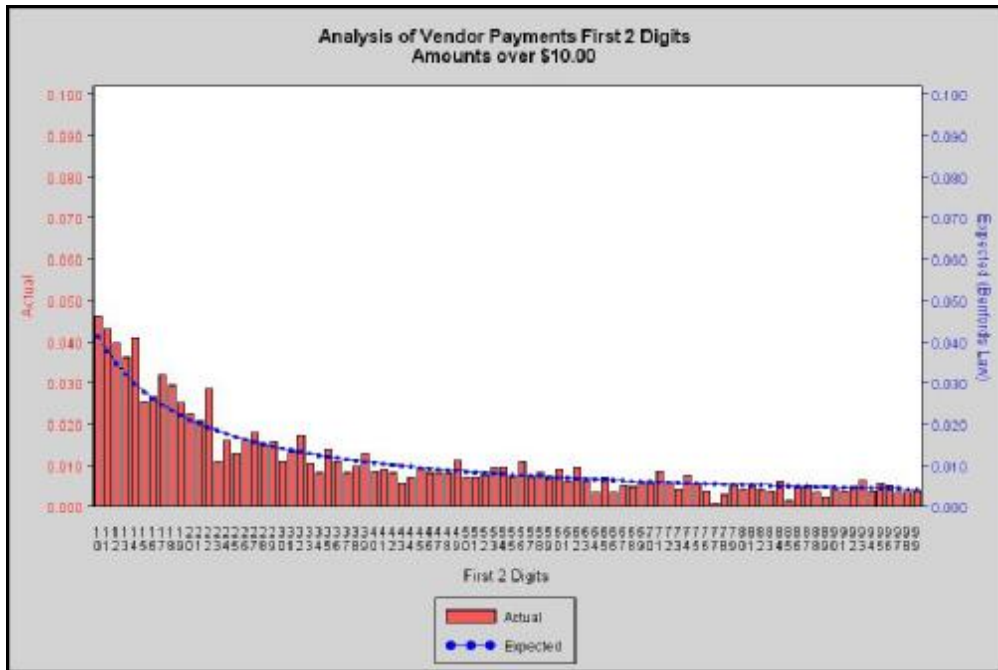


Figure A5. 34: Benford's Law – analysis of vendor payments

Benfords Law - Analysis of Vendor Payments
First 2 Digits = 22
 For Period: 01Dec03 to 31Dec03
 Number of Records Found = 60
[Back](#)
 User=1USRATTNC

Vendor Id	Doc. No.	Doc. Date	Post. Date	Amount	TCode
0000005157	1500096513	02DEC2003	02DEC2003	\$22.50	F110
0001002548	0500088527	02DEC2003	02DEC2003	\$222.22	F110
0001031603	0500088651	02DEC2003	02DEC2003	\$2,219.83	F110
0002000075	0500027835	02DEC2003	02DEC2003	\$2,259.01	F110
0001008118	1500096782	16DEC2003	16DEC2003	\$22.27	F110
0000010726	1500014022	16DEC2003	16DEC2003	\$221.20	F110
0001001988	0500028017	16DEC2003	16DEC2003	\$221.22	F110
0001003987	1500096766	16DEC2003	16DEC2003	\$221.83	F110
0001020101	0500089481	16DEC2003	16DEC2003	\$221.90	F110
0001004012	0500022869	16DEC2003	16DEC2003	\$225.00	F110
0000013613	0500089328	16DEC2003	16DEC2003	\$227.50	F110

Figure A5. 35: Benford's Law – investigation of spike at digit 22

Vendor Transaction History*
Transactions for Vendor: 0000009646
For Period: 01Dec03 to 31Dec03

Number of Records Found = 5
[View Users Interacting with this Vendor*](#)
[Back](#)

Date	User	TCode	Amount	Bank Details	Doc. No.
02/12/2003	1USRATTNC	F110	\$0.26		0500088697
03/12/2003	1USRINGGP	XK02	\$0.00	704-052439415	0001982140
04/12/2003	1USRCCRBD	F110	\$0.26		0500088697
04/12/2003	1USRATKSE	FK02	\$0.00	704-052439415S	0001982681
24/12/2003	1USRAGNKJ	FK02	\$0.00	704-052439415	0001998672

N = 5

Figure A5. 36: Transaction history for vendor – showing flipping

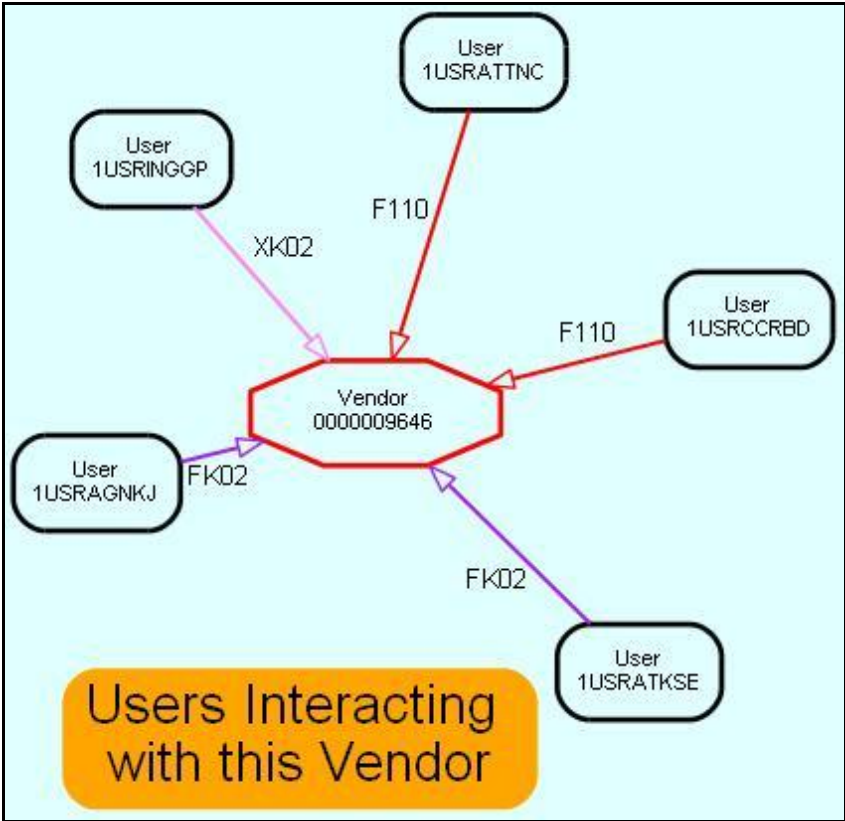


Figure A5. 37: Visualisation - users interacting with vendor

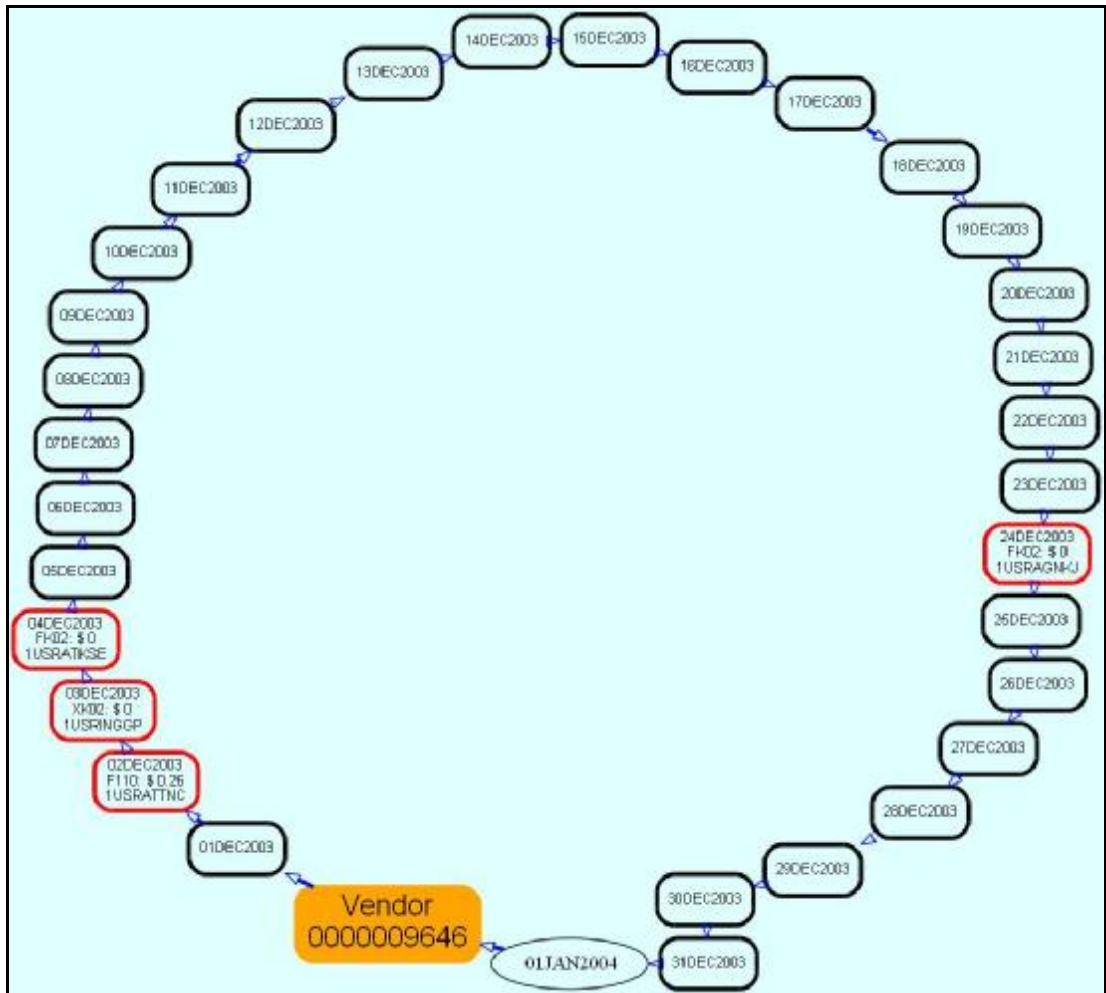


Figure A5. 38: Visualisation – vendor transaction history

Appendix 6: Results from case study 1a

Actual data from large international manufacturing company

Period of analysis : 01/01/2011 to 30/06/2011 (6 Months)

Analysis procedures

User profiles	Users are profiled to determine the scope of activities they have performed. Activities include vendor maintenance, invoicing and payment transactions. Summary and detailed reports are produced.
Critical combinations	Users that violate segregation of duties are identified and a report of potentially risky users is produced.
User activity analysis	An individual user is identified from the risky users list and selected for detailed investigation. Reports documenting individual user activities are produced.
Vendor analysis	A series of investigations are performed on active vendors, including vendors sharing bank accounts, vendors with multiple bank accounts, vendors with multiple master records, and Benford's law.

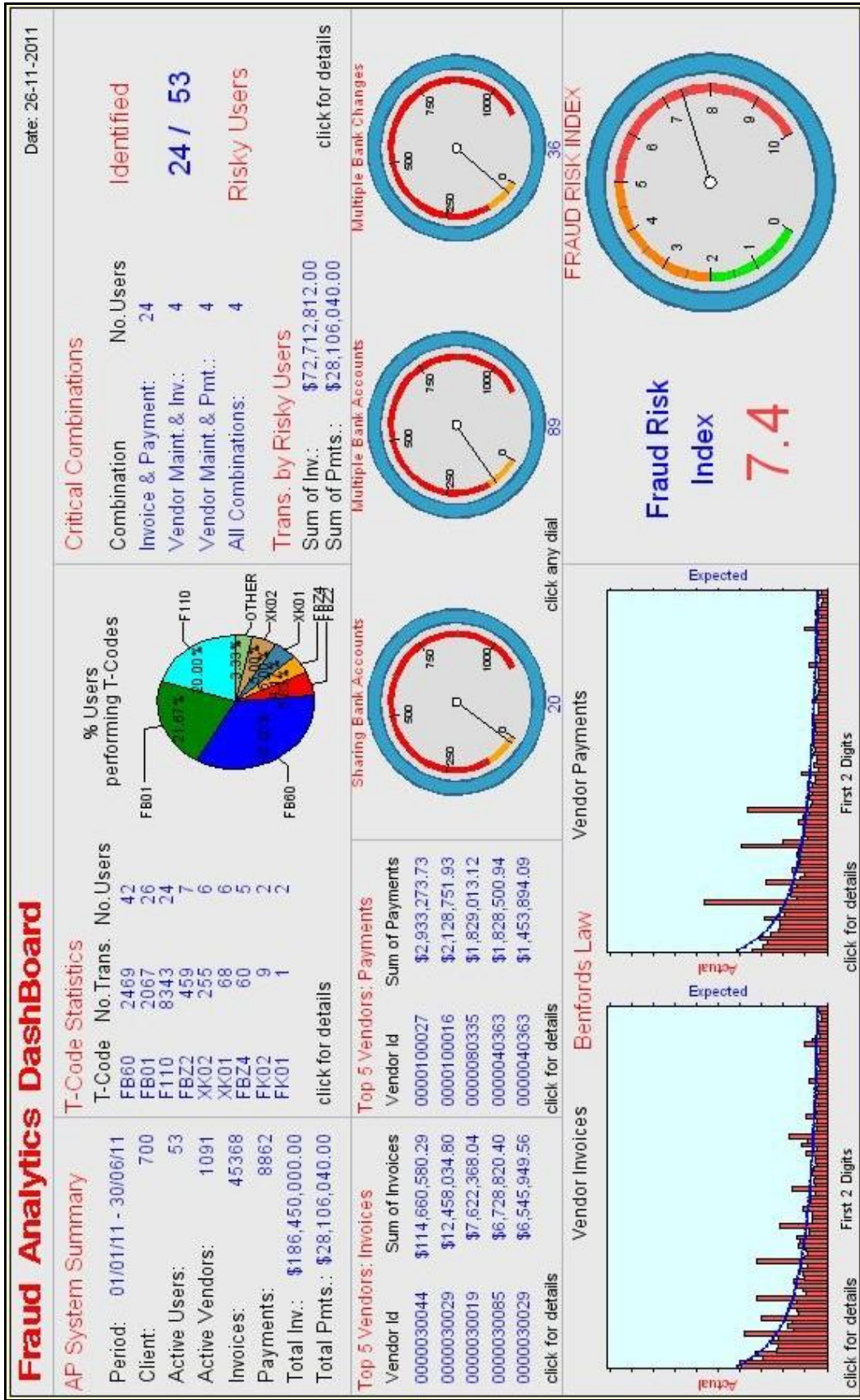


Figure A6.1: Dashboard

User profiling

User Activities Summary		
For Period: 01Jan11 to 30Jun11		
Number of Records Found = 9		
TCODE	Transaction Name	Activity
FB60	Enter Incoming Invoices	24690
FB01	Post Document	20678
F110	Parameters for Automatic Payment	8343
FBZ2	Post Outgoing Payments	459
XK02	Change vendor (centrally)	255
XK01	Create vendor (centrally)	68
FBZ4	Payment with Printout	60
FK02	Change Vendor (Accounting)	9
FK01	Create Vendor (Accounting)	1
N = 9		

Figure A6. 2: User activities summary

User Profile: Vendor Maintenance				
For Period: 01Jan11 to 30Jun11				
Number of Records Found = 10				
User	Transaction Code			
	FK01	FK02	XK01	XK02
1USRN	.	.	66	197
1USRMI	.	.	1	36
1USRA	.	9	.	2
1USREEWAH	.	.	.	17
1USRRAGU	.	.	.	3
1USRGL-POW	.	.	1	.
1USRPOWER	1	.	.	.
TOTAL	1	9	68	255

Figure A6. 3: User profile – vendor maintenance

User Profile: Invoice Transactions
For Period: 01Jan11 to 30Jun11
Number of Records Found = 46

User	Transaction Code	
	FB60	FB01
1USRTHRI	414	188
1USRTHAW	487	26
1USRINDUD	92	29
1USRSH	5188	7613
1USRHANI	2734	3219
1USRSHA	2477	277
1USRSHIKA	2808	1179
1USREAM	24	566
1USRA	101	162

Figure A6. 4: User profile – invoice transactions

User Profile: Payment Transactions
For Period: 01Jan11 to 30Jun11
Number of Records Found = 24

User	Transaction Code		
	FBZ4	FBZ2	F110
1USRSHIKA	7	144	1224
1USRHANI	43	48	1311
1USRSHA	2	179	970
1USRMI	1	78	241
1USRALIN	7	3	1517
1USRDU	-	6	1072
1USRINDUD	-	-	153
1USREWA	-	-	68
1USRA	-	1	17

Figure A6. 5: User profile – payment transactions

User Profile: Invoices or Payment Transactions
For Period: 01Jan11 to 30Jun11
Number of Records Found = 70

User	Transaction Code				
	FBZ4	FBZ2	FB60	FB01	F110
1USRSHIKA	7	144	2808	1179	1224
1USRHANI	43	48	2734	3219	1311
1USRSHA	2	179	2477	277	970
1USRMI	1	78	720	17	241
1USRINDUD	.	.	92	29	153
1USRALIN	7	3	3005	1918	1517
1USRA	.	1	101	162	17
1USRSH	.	.	5188	7613	1456
1USRDU	.	6	78	2	1072

Figure A6. 6: User profile – invoices or payment transactions

User Profile: All Combinations*
For Period: 01Jan11 to 30Jun11
Number of Records Found = 78

User	Transaction Code								
	XK02	XK01	FK02	FK01	FBZ4	FBZ2	FB60	FB01	F110
1USRSHIKA	7	144	2808	1179	1224
1USRHANI	43	48	2734	3219	1311
1USRSHA	2	179	2477	277	970
1USRMI	36	1	.	.	1	78	720	17	241
1USRA	2	.	9	.	.	1	101	162	17
1USRINDUD	92	29	153
1USRALIN	7	3	3005	1918	1517
1USRN	197	66	13	13	14
1USRSH	5188	7613	1456

Figure A6. 7: User profile – all combinations

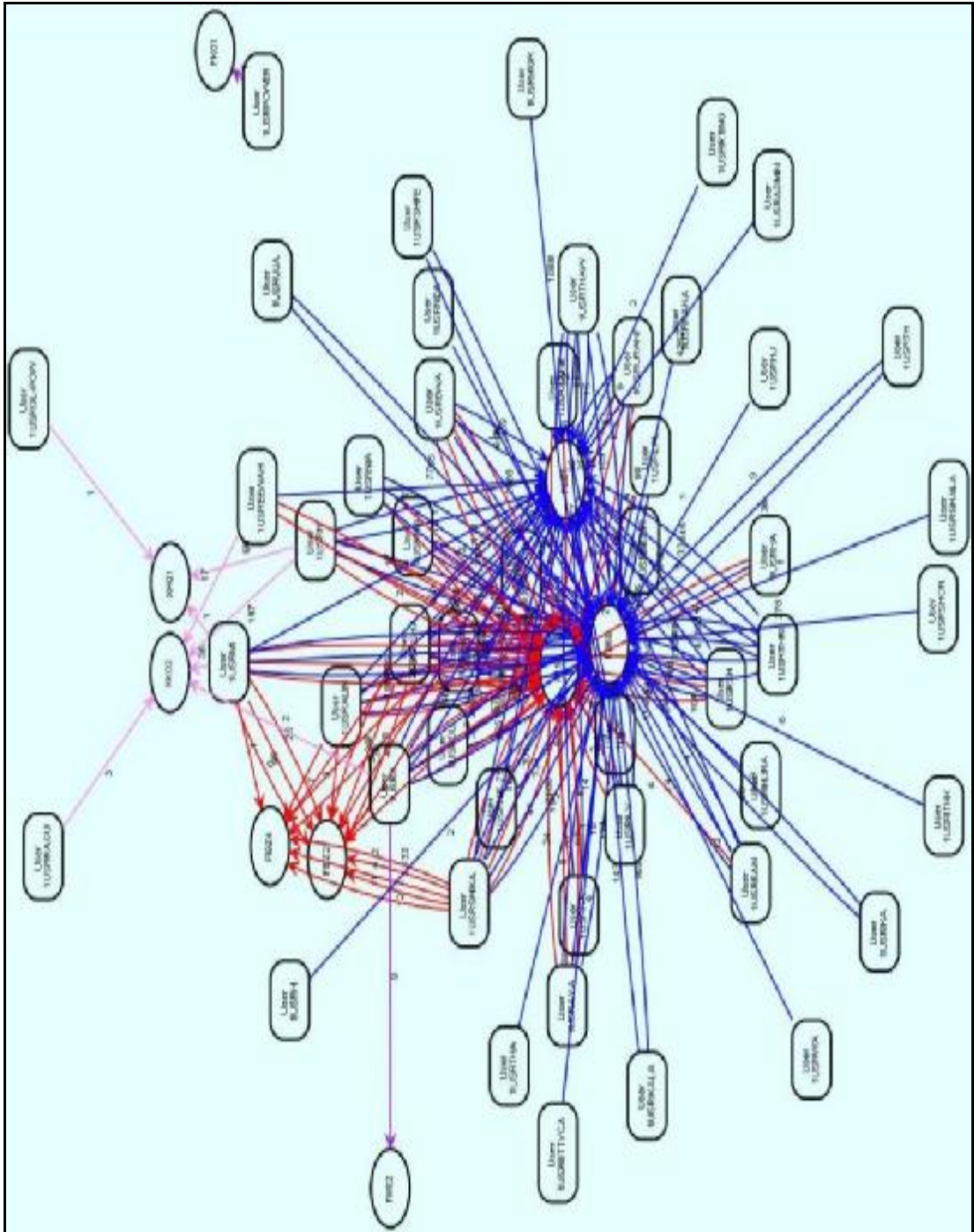


Figure A6. 8: Visualisation – all combinations

Critical combinations

Breach of Segregation of Duties
Users Entering Invoices and Payments*

For Period: 01Jan11 to 30Jun11

Number of Records Found = 66

User	Enter Invoice	Enter Payment
1USR	FB01	F110
1USR	FB60	F110
1USRA	FB01	F110
1USRA	FB01	FB22
1USRA	FB60	F110
1USRA	FB60	FB22
1USRALIN	FB01	F110
1USRALIN	FB01	FB22
1USRALIN	FB01	FB24
1USRALIN	FB60	F110

Figure A6. 9: Violation of SoDs – users entering invoices and payments

Breach of Segregation of Duties
Users Performing Vendor Maintenance and Entering Invoices*

For Period: 01Jan11 to 30Jun11

Number of Records Found = 13

User	Vendor Maint.	Enter Invoice
1USRA	FK02	FB01
1USRA	XK02	FB01
1USRA	FK02	FB60
1USRA	XK02	FB60
1USREEWAH	XK02	FB01
1USRMI	XK01	FB01
1USRMI	XK02	FB01
1USRMI	XK01	FB60
1USRMI	XK02	FB60

Figure A6. 10: Violation of SoDs – users performing vendor maintenance and entering invoices

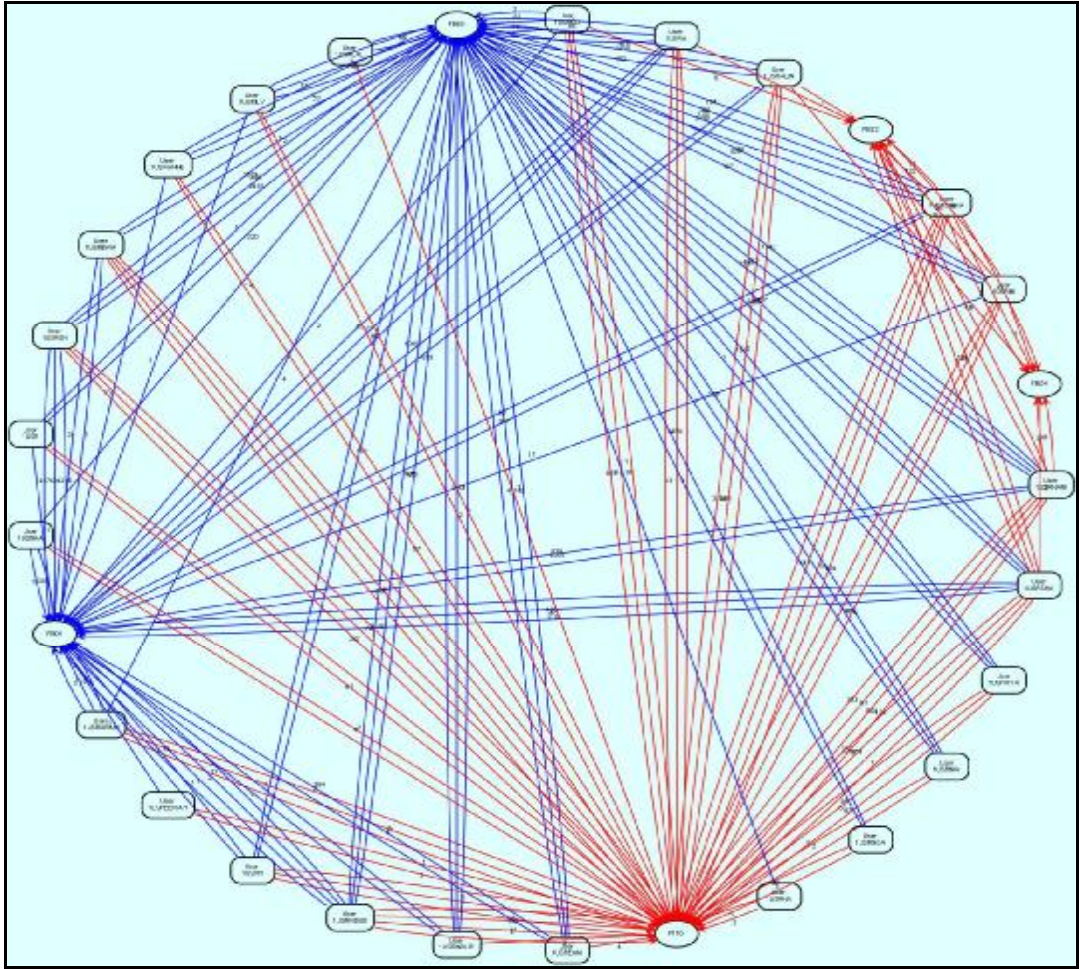


Figure A6. 11: Visualisation - users entering invoices and payments

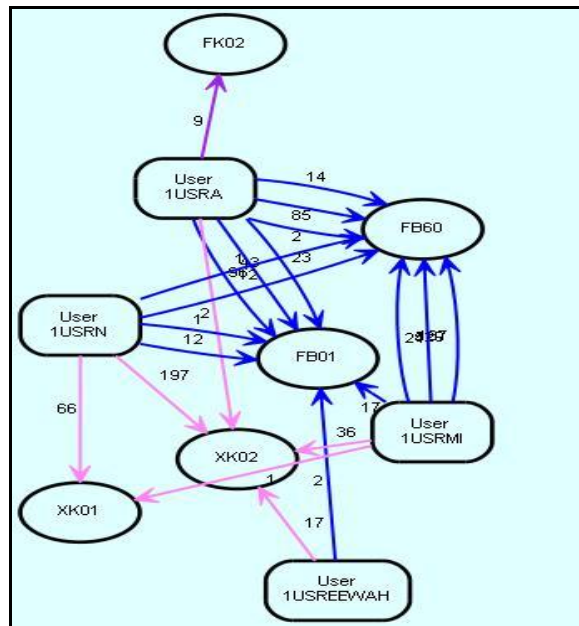


Figure A6. 12: Visualisation - users performing vendor maintenance and entering invoices

Breach of Segregation of Duties
Users Performing Vendor Maintenance and Entering Payments*

For Period: 01Jan11 to 30Jun11

Number of Records Found = 13

User	Vendor Maint.	Enter Payment
1USRA	FK02	F110
1USRA	XK02	F110
1USRA	FK02	FBZ2
1USRA	XK02	FBZ2
1USREEWAH	XK02	F110
1USRMI	XK01	F110
1USRMI	XK02	F110
1USRMI	XK01	FBZ2
1USRMI	XK02	FBZ2

Figure A6. 13: Violation of SoDs – users performing vendor maintenance and entering payments

Breach of Segregation of Duties
Users Performing Vendor Maintenance, Entering Invoices and Payments*

For Period: 01Jan11 to 30Jun11

Number of Records Found = 25

User	Vendor Maint.	Enter Invoice	Enter Payment
1USRA	FK02	FB01	F110
1USRA	XK02	FB01	F110
1USRA	FK02	FB01	FBZ2
1USRA	XK02	FB01	FBZ2
1USRA	FK02	FB60	F110
1USRA	XK02	FB60	F110
1USRA	FK02	FB60	FBZ2
1USRA	XK02	FB60	FBZ2
1USREEWAH	XK02	FB01	F110

Figure A6. 14: Violation of SoDs – users performing vendor maintenance, entering invoices and payments

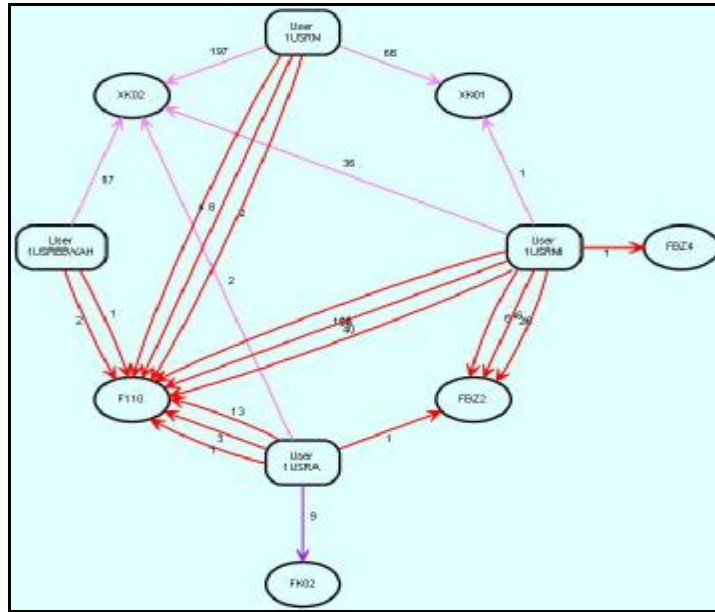


Figure A6. 15: Visualisation - users performing vendor maintenance and entering payments

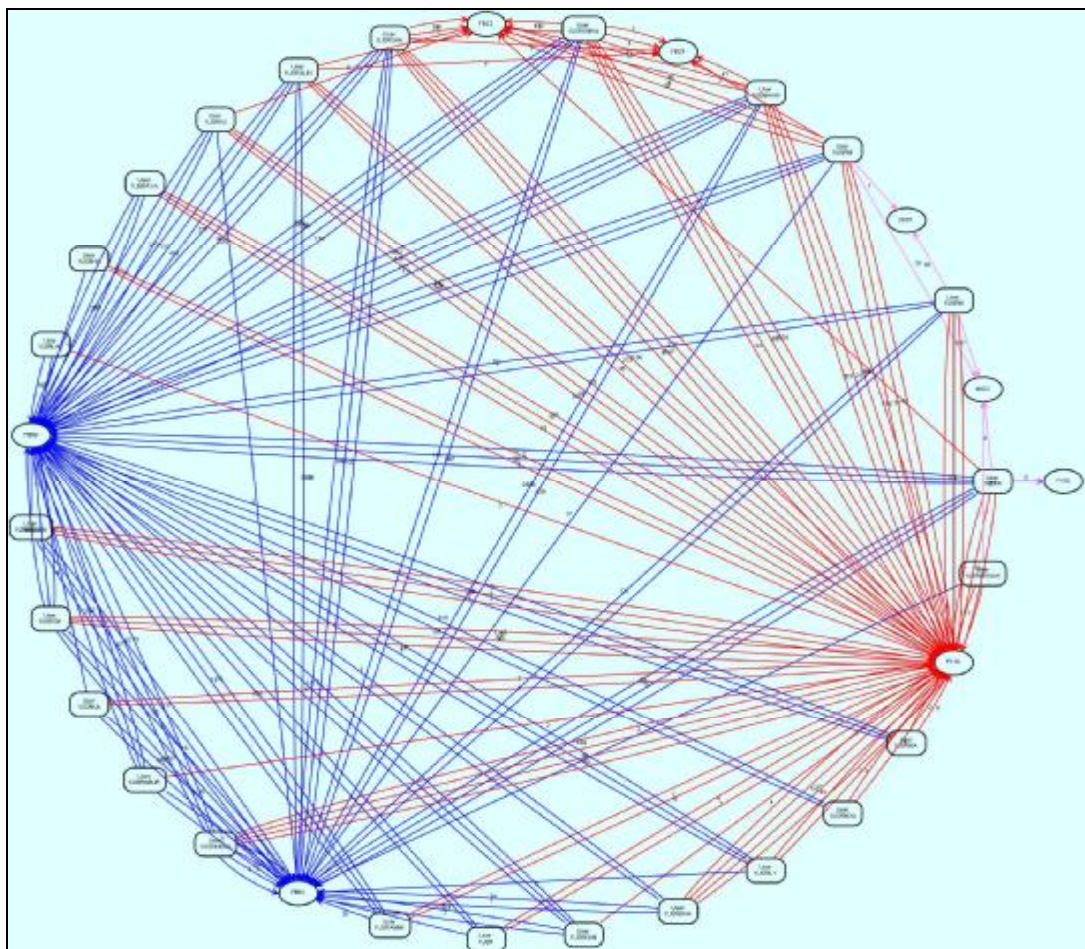


Figure A6. 16: Visualisation - users performing vendor maintenance, entering invoices and payments

Vendor analysis

Vendor Id		Vendor
0000021580	2VENDORERMANN WIRKWARENERZEUGUNG	
0000050110	2VENDORERMANN WIRKWARENERZEUGUNG	
N = 2		

Vendor Id		Vendor
0000041319	2VENDORINTING & PACKAGING (PVT) LTD	
0000080079	2VENDORINTING & PACKAGING (PVT) LTD	
N = 2		

Figure A6. 17: Vendors sharing bank accounts

Vendors with Multiple Bank Accounts*					
For Period: 01Jan11 to 30Jun11					
Number of Records Found = 89					
Vendor Id=0000010071					
Vendor Name	Bank Details	Date	Time	User	
2VENDORIONS LABELS LANKA (PVT) LTD	3BANKBH 011026960101	01/02/11	11:35:46	1USRN	
2VENDORIONS LABELS LANKA (PVT) LTD	3BANKM 9911090796001	04/01/11	10:10:29	1USRN	
N = 2					
Vendor Id=0000010103					
Vendor Name	Bank Details	Date	Time	User	
2VENDORLANKA PRINTS (PVT) LIMITED	3BANKKCM 0070631839	13/01/11	12:54:45	1USRN	
2VENDORLANKA PRINTS (PVT) LIMITED	3BANKKCM 70631839	06/01/11	10:12:45	1USRN	
N = 2					

Figure A6. 18: Vendors with multiple bank accounts

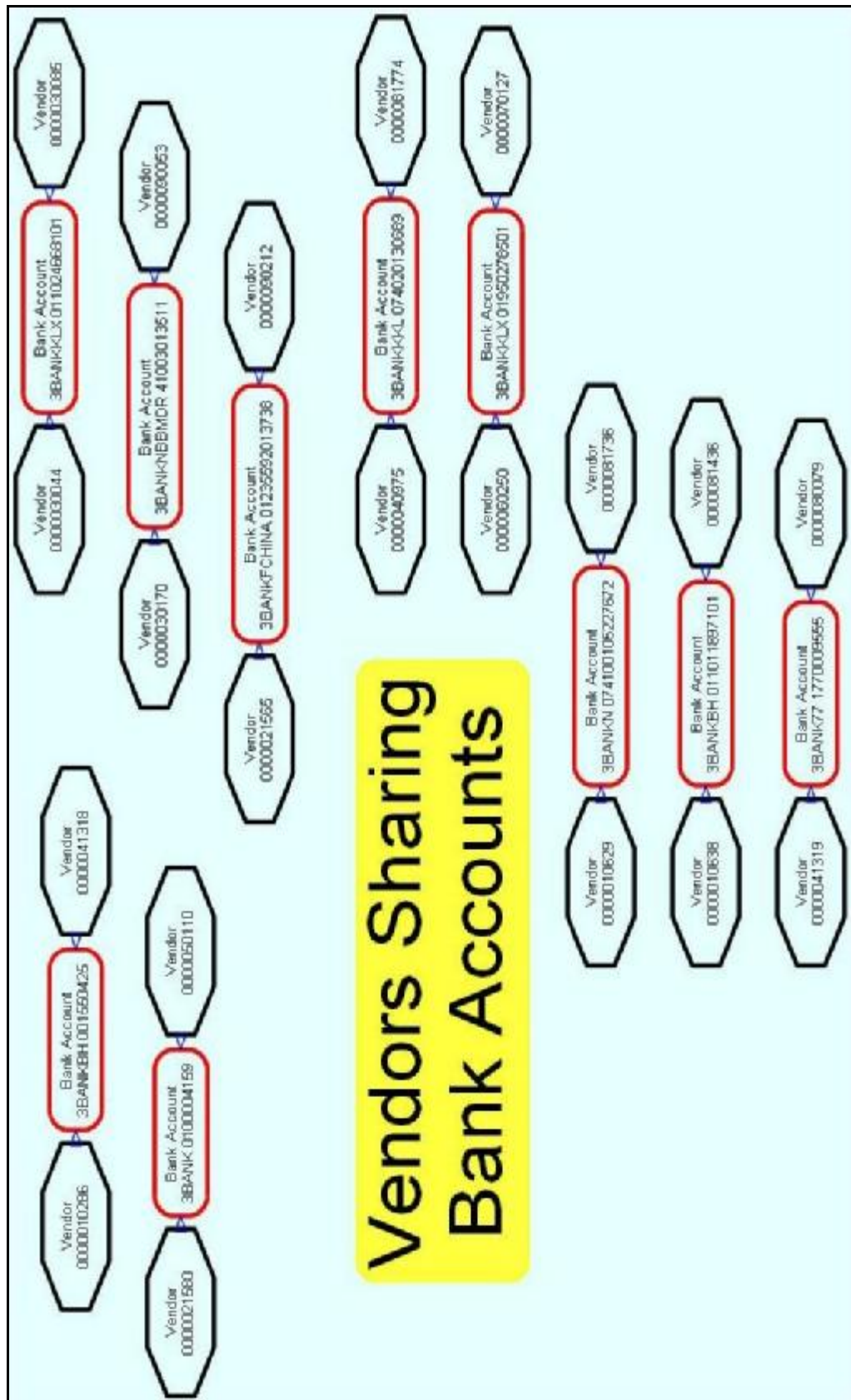


Figure A6. 19: Visualisation - vendors sharing bank accounts

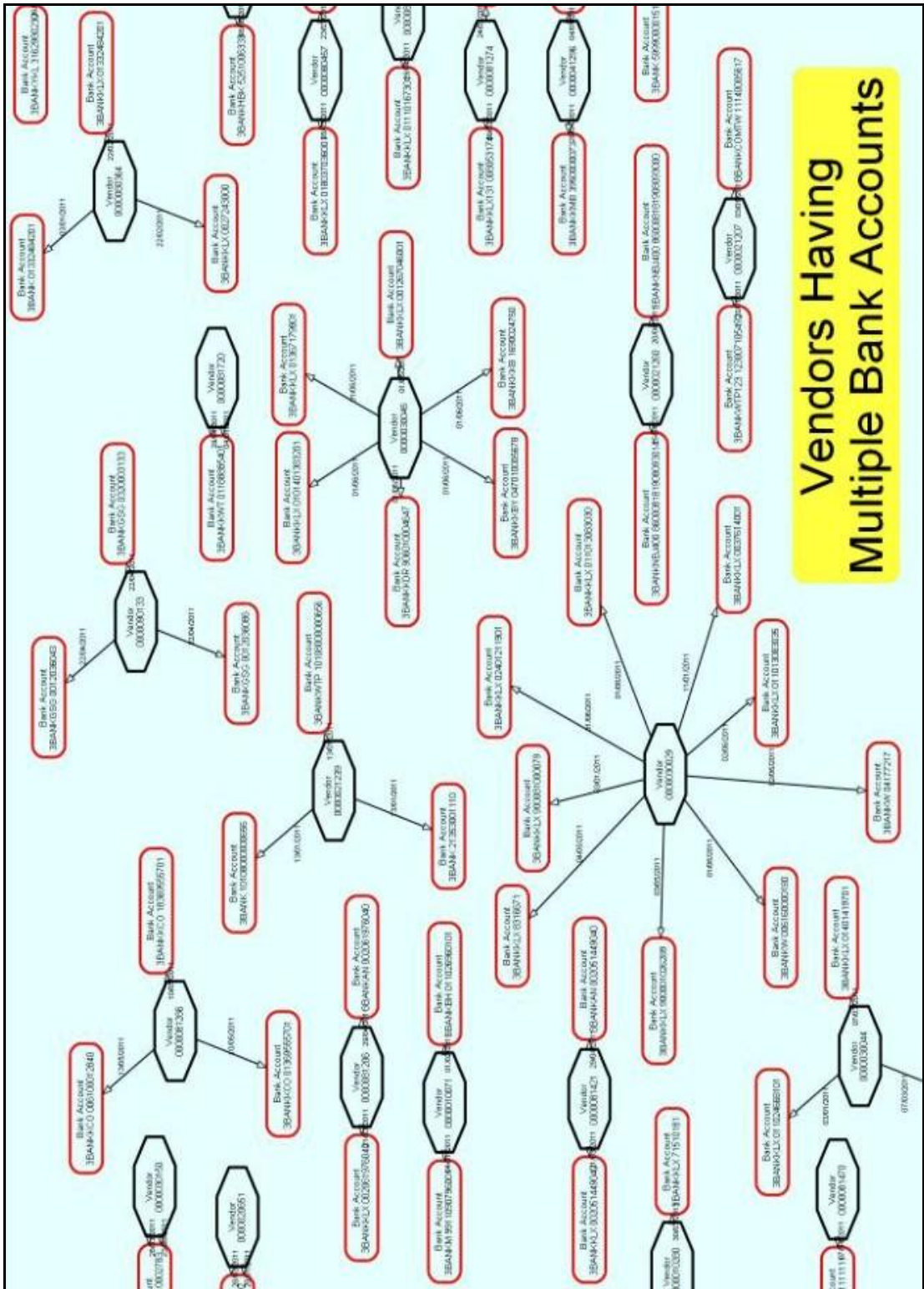


Figure A6. 20: Visualisation - vendors having multiple bank accounts

Vendors: Multiple Bank Changes		
For Period: 01Jan11 to 30Jun11		
Number of Records Found = 36		
Vendor Id	Vendor Name	No of Changes
0000030029	2VENDORITAL (PVT) LTD	9
0000030045	2VENDOR (PVT) LTD	6
0000010186	2VENDOR(PVT) LIMITED	3
0000021239	2VENDOR TEXTILE CO , LTD	3
0000030044	2VENDORA (PVT) LTD - (TRAD PAY)	3
0000080364	2VENDORLTD	3
0000081286	2VENDOR(PVT) LTD	3
0000090133	2VENDORL UNIVERSITY OF SINGAPORE	3
0000010071	2VENDORIONS LABELS LANKA (PVT) LTD	2
0000010103	2VENDORLANKA PRINTS (PVT) LIMITED	2
0000010280	2VENDOR EMBROIDERY SOLUTION (PVT)	2

Figure A6. 21: Vendors with multiple changes to their bank accounts

Vendors: Multiple Master Records	
Number of Records Found = 393	
Vendor	No. of Master Records
2VENDOR	68
2VENDOR (PVT) LTD	14
2VENDOR(PVT) LTD	9
2VENDORTERPRISES	9
2VENDORNTERPRISES	8
2VENDOR Enterprises	7
2VENDORnterprises	7
2VENDORE (PVT) LTD	6
2VENDORENTERPRISES	6
2VENDORe	6
2VENDOR ENTERPRISES	5

Figure A6. 22: Vendors with multiple master records

TOP 5 Vendors: Sum of Invoices				
For Period: 01Jan11 to 30Jun11				
Number of Records Found = 5				
Co Code	Vendor Id	Vendor	Sum of Invoices	No of Invoices
6000	0000030044	2VENDORA (PVT) LTD - (TRAD PAY)	\$114,660,580.29	5269
3000	0000030029	2VENDORITAL (PVT) LTD	\$12,458,034.80	54
3000	0000030019	2VENDORA (PVT) LTD	\$7,622,368.04	339
6000	0000030085	2VENDORIMATES THURULIE (PVT) LTD	\$6,728,820.40	538
6000	0000030029	2VENDORITAL (PVT) LTD	\$6,546,949.56	41
			\$148,015,753.09	
N = 5				

Figure A6. 23: Top 5 vendors by sum of invoices

TOP 5 Vendors: Sum of Payments				
For Period: 01Jan11 to 30Jun11				
Number of Records Found = 5				
Co Code	Vendor Id	Vendor	Sum of Payments	No of Payments
3000	0000100027	2VENDOR	\$2,933,273.73	291
3000	0000100016	2VENDORTENDENT EMPLOYEES	\$2,128,751.93	118
3000	0000080335	2VENDORAGER COMMERCIAL BANK -	\$1,829,013.12	74
6000	0000040363	2VENDORSSURANCE PLC	\$1,828,500.94	90
3000	0000040363	2VENDORSSURANCE PLC	\$1,463,894.09	47
			\$10,173,433.81	
N = 5				

Figure A6. 24: Top 5 vendors by sum of payments

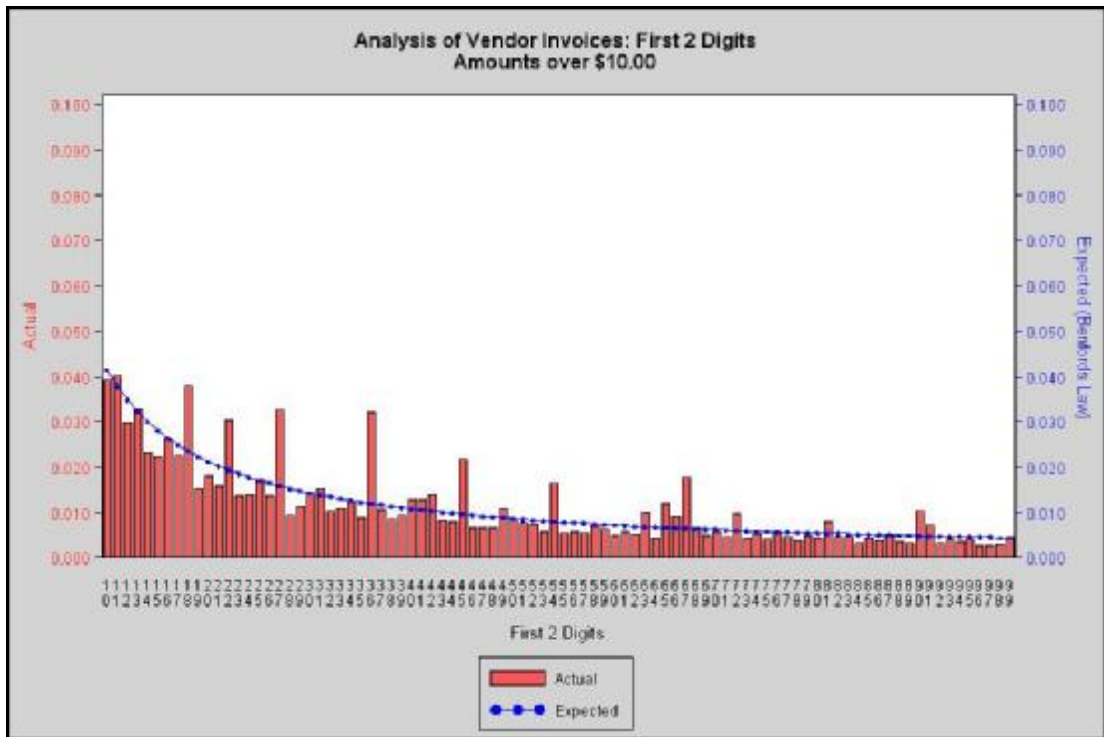


Figure A6. 25: Benford's Law – analysis of vendor invoices

Benfords Law - Analysis of Vendor Invoices
First 2 Digits = 36
 For Period: 01Jan11 to 30Jun11
 Number of Records Found = 1217
[Back](#)
 User=1USR

Vendor Id	Doc. No.	Doc. Date	Post. Date	Amount	TCode
0000081163	1900082242	26JAN2011	26JAN2011	\$36.05	FB60
0000081163	1900082949	31JAN2011	31JAN2011	\$36.05	FB60
0000081163	1900083977	10FEB2011	10FEB2011	\$36.08	FB60
0000081163	1900085694	28FEB2011	28FEB2011	\$36.08	FB60
0000081163	1900086224	03MAR2011	03MAR2011	\$36.11	FB60
0000081163	1900086840	10MAR2011	10MAR2011	\$36.11	FB60
0000081163	1900086846	10MAR2011	10MAR2011	\$36.11	FB60
0000081163	1900087693	21MAR2011	21MAR2011	\$36.11	FB60

Figure A6. 26: Benford's Law – investigation of spike at digit 36

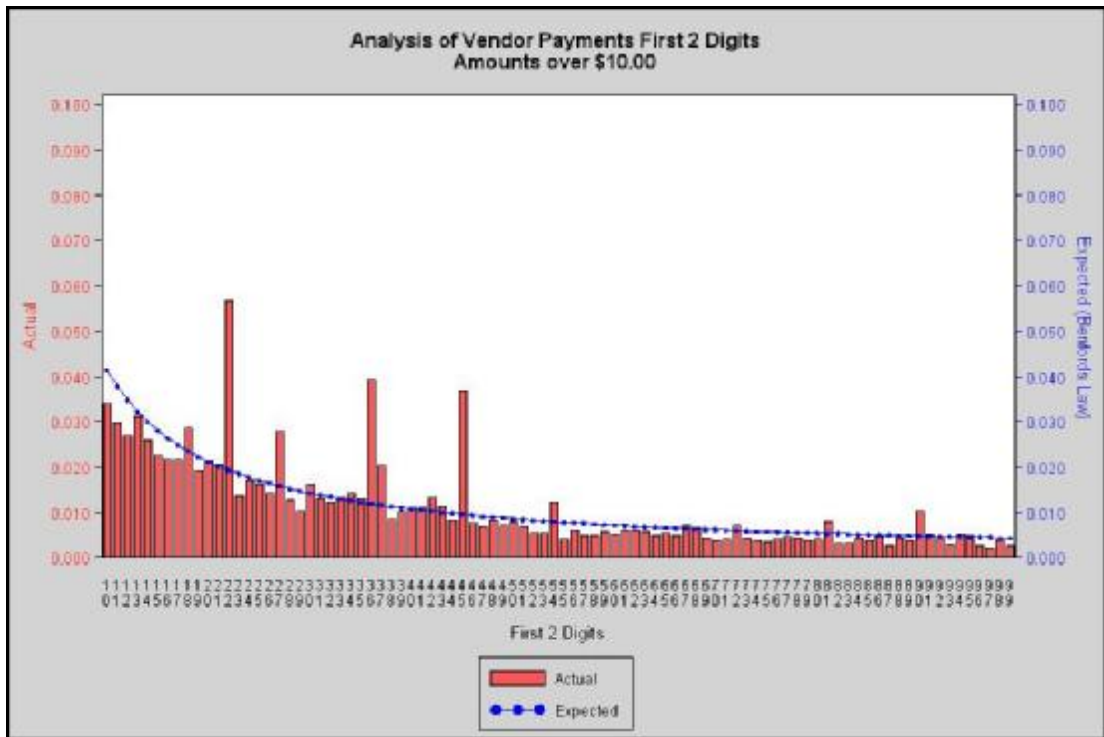


Figure A6. 27: Benford's Law – analysis of vendor payments

Benfords Law - Analysis of Vendor Payments
First 2 Digits = 22
 For Period: 01Jan11 to 30Jun11
 Number of Records Found = 499
[Back](#)
 User=1USRALIN

Vendor Id	Doc. No.	Doc. Date	Post. Date	Amount	TCode
0000080168	1500031874	05JAN2011	05JAN2011	\$22.47	F110
0000040120	1500031861	05JAN2011	05JAN2011	\$222.67	F110
0000080068	1500032186	13JAN2011	13JAN2011	\$22,532.67	F110
0000041000	1500032230	17JAN2011	17JAN2011	\$227.13	F110
0000060116	1500032222	17JAN2011	17JAN2011	\$228.03	F110
0000081347	1500023708	20JAN2011	20JAN2011	\$2,273.62	F110
0000080168	1500032358	21JAN2011	21JAN2011	\$229.55	F110
0000080363	1500032363	21JAN2011	21JAN2011	\$2,208.75	F110

Figure A6. 28: Benford's Law – investigation of spike at digit 22

Appendix 7: Results from case study 1b

Actual data from large international manufacturing company

Period of analysis : 01/06/2011 to 07/06/2011 (7 days)

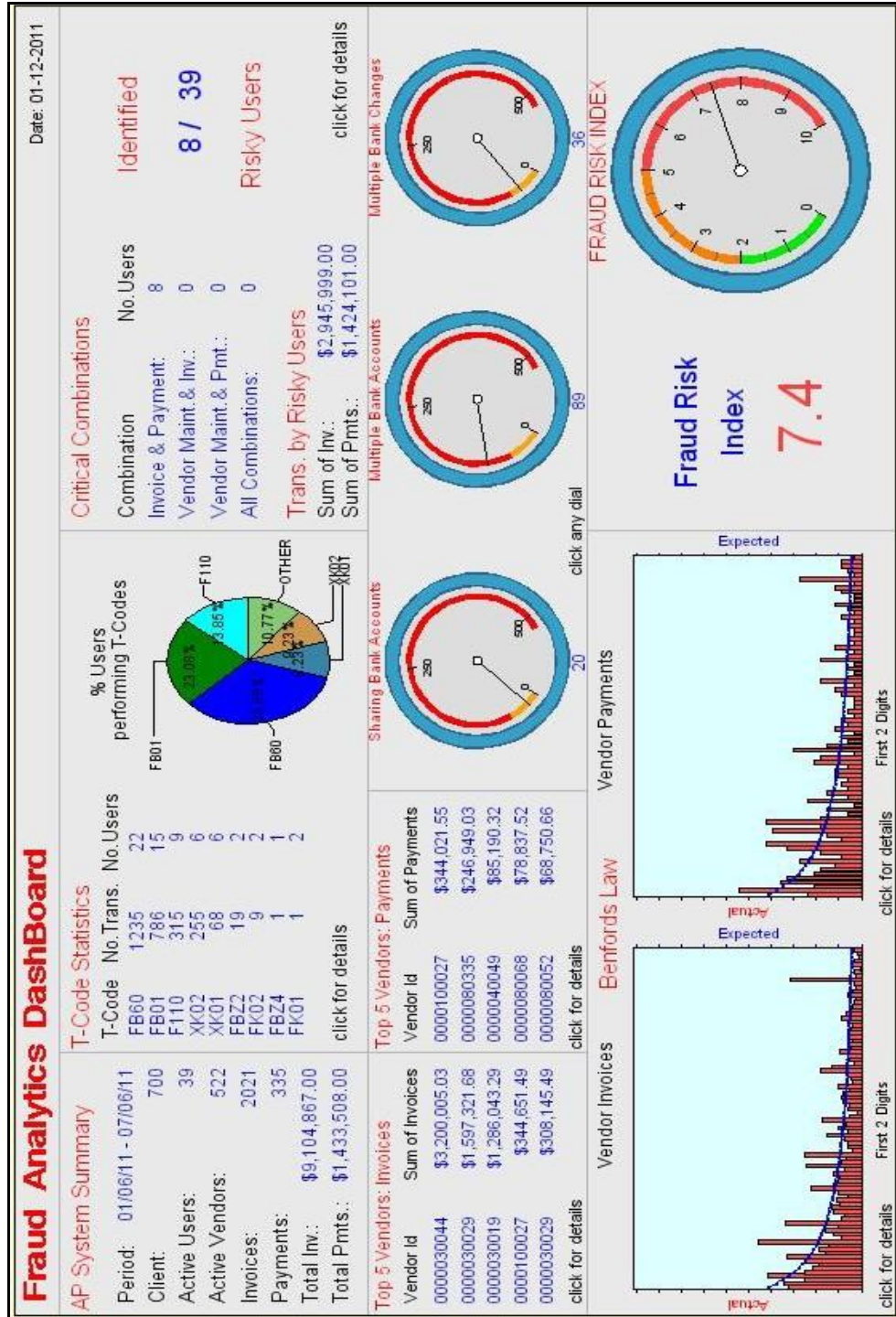


Figure A7.1: Dashboard

User Activities Summary
For Period: 01Jun11 to 07Jun11
Number of Records Found = 9

TCODE	Transaction Name	Activity
FB60	Enter Incoming Invoices	1235
FB01	Post Document	786
F110	Parameters for Automatic Payment	315
XK02	Change vendor (centrally)	255
XK01	Create vendor (centrally)	68
FBZ2	Post Outgoing Payments	19
FK02	Change Vendor (Accounting)	9
FK01	Create Vendor (Accounting)	1
FBZ4	Payment with Printout	1
N = 9		

Figure A7. 2: User activities summary

Breach of Segregation of Duties
Risky User List
For Period: 01Jun11 to 07Jun11
Number of Records Found = 11

User	Invoice & Payment	Vendor Maint & Invoice	Vendor Maint & Payment	Vendor Maint & Invoice & Payment
1USRA		>X<		
1USRALIN	>X<			
1USRDU	>X<			
1USREWA	>X<			
1USRHANI	>X<			
1USRINDUD	>X<			
1USRMI		>X<		
1USRN		>X<		
1USRSH	>X<			

Figure A7. 3: Risky user list

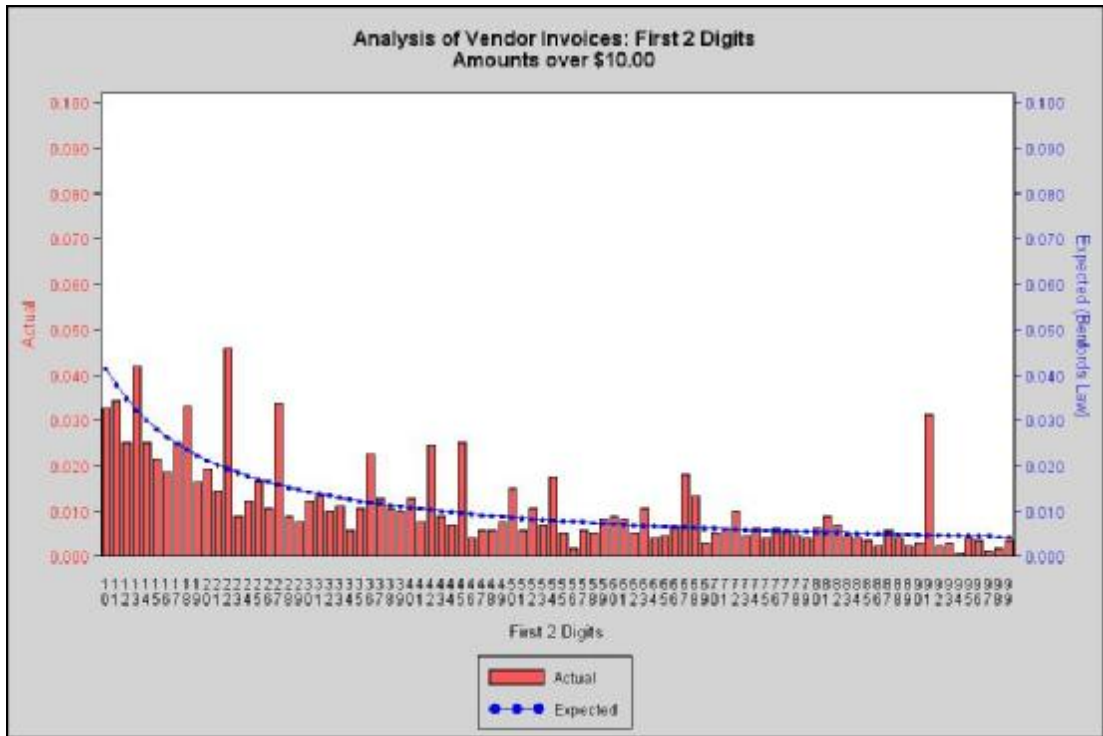


Figure A7. 4: Benford's Law – analysis of vendor invoices

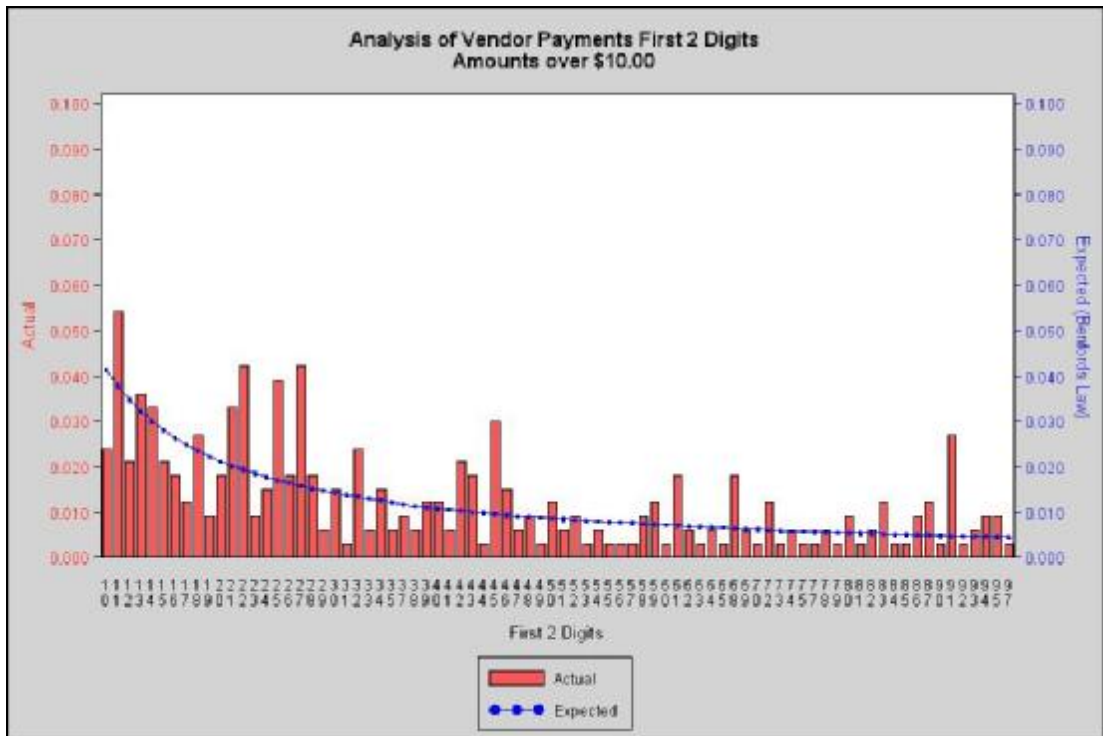


Figure A7. 5: Benford's Law – analysis of vendor payments

Appendix 8: Data extraction

Tables and fields required for data extraction

Table A8.1: SAP tables and field requirements

CDHDR	
MANDANT	Client
OBJECTCLAS	Change doc. Object
OBJECTID	Object Value
CHANGENR	Document Number
USERNAME	Name
UDATE	Date
UTIME	Time
TCODE	Transaction Code

CDPOS	
MANDANT	Client
OBJECTCLAS	Change doc. object
OBJECTID	Object value
CHANGENR	Document Number
TABNAME	Table Name
TABKEY	Table Key
FNAME	Field Name
CHNGIND	Change ID


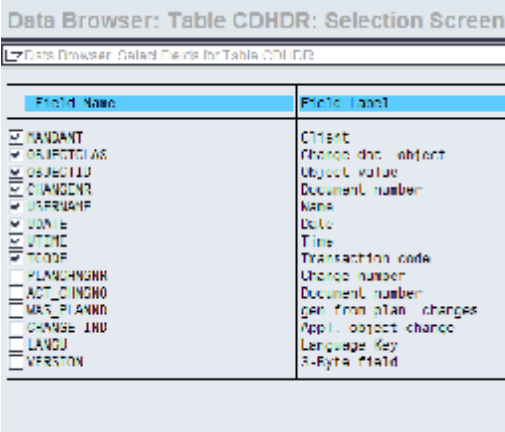
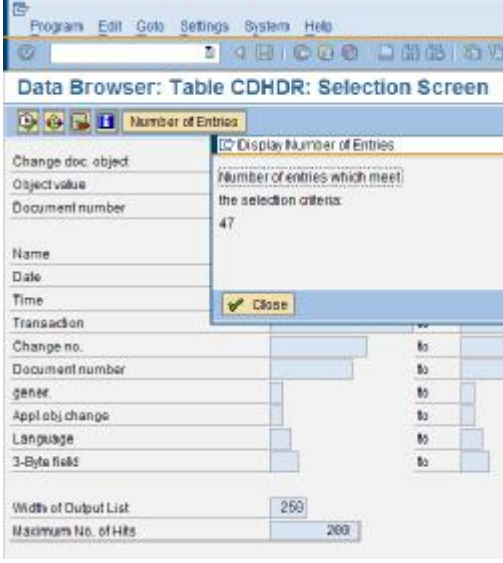
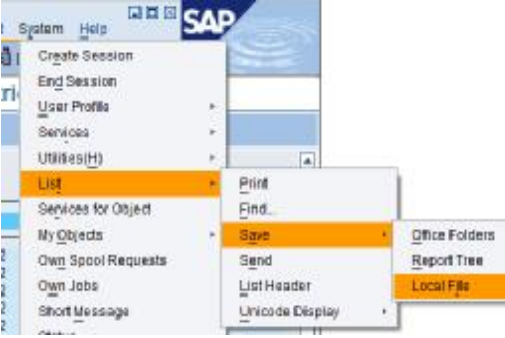
BKPF	
MANDANT	Client
BUKRS	Company Code
BELNR	Document Number
GJAHR	Fiscal Year
BLART	Document Type
BLDAT	Document Date
BUDAT	Posting Date
MONAT	Posting Period
CPUDT	Entry Date
CPUTM	Entry Time
USNAM	User Name
TCODE	Transaction Code

BSEG	
MANDANT	Client
BUKRS	Company Code
BELNR	Document Number
GJAHR	Fiscal Year
AUGBL	Clearing Document Number
BSCHL	Posting Key
SHKZG	Debit / Credit Indicator
WRBTR	Amount
HKONT	G/L Account
KUNNR	Customer Number
LIFNR	Vendor Number

LFA1	
MANDT	Client
LIFNR	Vendor Number
ERDAT	Created On
ERNAM	Created By
NAME1	Name

Summary extraction process

Table A8.2: Summary data extraction procedure

<p>Enter table name for extraction and click on Enter.</p> 	<p>In the following screen click on Settings > Format List > Choose Fields. Deselect all fields and just select the required fields. Click on Transfer.</p>  <table border="1" data-bbox="895 786 1402 1077"> <thead> <tr> <th>Field Name</th> <th>Field Label</th> </tr> </thead> <tbody> <tr><td><input type="checkbox"/> RANDANT</td><td>Client</td></tr> <tr><td><input type="checkbox"/> OBJECTCLAS</td><td>Change doc. object</td></tr> <tr><td><input type="checkbox"/> OBJECTID</td><td>Object value</td></tr> <tr><td><input type="checkbox"/> CHANGENR</td><td>Document number</td></tr> <tr><td><input type="checkbox"/> USERPRINF</td><td>Name</td></tr> <tr><td><input type="checkbox"/> DATE</td><td>Date</td></tr> <tr><td><input type="checkbox"/> TIME</td><td>Time</td></tr> <tr><td><input type="checkbox"/> TCODE</td><td>Transaction code</td></tr> <tr><td><input type="checkbox"/> PLANCHNGR</td><td>Change number</td></tr> <tr><td><input type="checkbox"/> ACT_PLNCHG</td><td>Document number</td></tr> <tr><td><input type="checkbox"/> MAS_PLANNR</td><td>gen. from plan changes</td></tr> <tr><td><input type="checkbox"/> CHANGE_IND</td><td>Appl. object change</td></tr> <tr><td><input type="checkbox"/> LANGU</td><td>Language Key</td></tr> <tr><td><input type="checkbox"/> VERSTW</td><td>3-Byte field</td></tr> </tbody> </table>	Field Name	Field Label	<input type="checkbox"/> RANDANT	Client	<input type="checkbox"/> OBJECTCLAS	Change doc. object	<input type="checkbox"/> OBJECTID	Object value	<input type="checkbox"/> CHANGENR	Document number	<input type="checkbox"/> USERPRINF	Name	<input type="checkbox"/> DATE	Date	<input type="checkbox"/> TIME	Time	<input type="checkbox"/> TCODE	Transaction code	<input type="checkbox"/> PLANCHNGR	Change number	<input type="checkbox"/> ACT_PLNCHG	Document number	<input type="checkbox"/> MAS_PLANNR	gen. from plan changes	<input type="checkbox"/> CHANGE_IND	Appl. object change	<input type="checkbox"/> LANGU	Language Key	<input type="checkbox"/> VERSTW	3-Byte field
Field Name	Field Label																														
<input type="checkbox"/> RANDANT	Client																														
<input type="checkbox"/> OBJECTCLAS	Change doc. object																														
<input type="checkbox"/> OBJECTID	Object value																														
<input type="checkbox"/> CHANGENR	Document number																														
<input type="checkbox"/> USERPRINF	Name																														
<input type="checkbox"/> DATE	Date																														
<input type="checkbox"/> TIME	Time																														
<input type="checkbox"/> TCODE	Transaction code																														
<input type="checkbox"/> PLANCHNGR	Change number																														
<input type="checkbox"/> ACT_PLNCHG	Document number																														
<input type="checkbox"/> MAS_PLANNR	gen. from plan changes																														
<input type="checkbox"/> CHANGE_IND	Appl. object change																														
<input type="checkbox"/> LANGU	Language Key																														
<input type="checkbox"/> VERSTW	3-Byte field																														
<p>In the following screen check Number of Entries. Set Maximum No. of Hits (default is 200). Click Execute.</p> 	<p>Export the file for external analysis. Click System > List > Save > Local File.</p> 																														

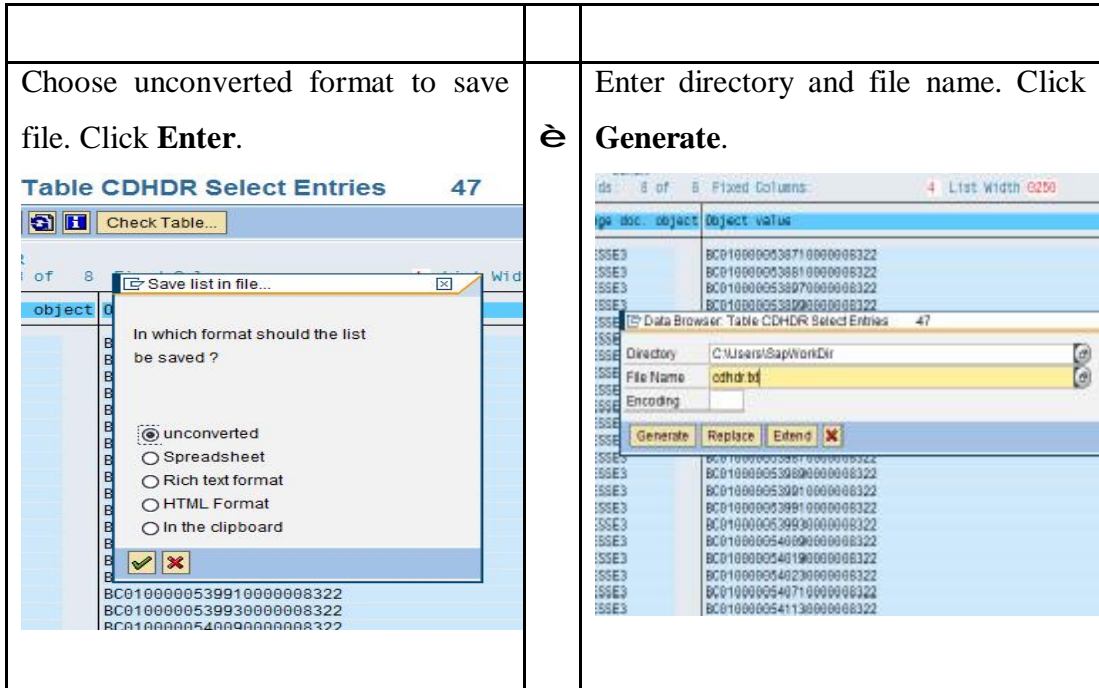


Table extraction

Table A8.3: SAP table extraction documentation

Step	Table	Procedure	Description	Complete	Sign.
1	BKPF	<p>SE16 Enter table name</p> <p>Display Table Contents (Enter)</p> <p>Settings > Format List > Choose Fields Select fields (<i>as shown in field requirements, section 7.2</i>)</p> <p>Check Number of Entries Set Maximum No. of Hits</p> <p>Enter filter parameters, e.g. Document Date, Fiscal Year (If a specific field is not</p>	Accounting Document Header	..	_____

Step	Table	Procedure	Description	Complete	Sign.
		<p>present it may be added to the selection screen by clicking on Settings > Fields for Selection)</p> <p>Execute</p> <p>System > List > Save > Local File > Unconverted Choose Directory Enter Filename: BKPF.TXT</p>			
2	BSEG	<p>As above (step 1) Enter filter parameters, e.g. Fiscal Year.</p> <p>Enter Filename: BSEG.TXT</p>	Accounting document Line Item	..	_____
3	CDHDR	<p>As above (step 1) Enter filter parameters, e.g. Date.</p> <p>Enter Filename: CDHDR.TXT</p>	Change Document Header	..	_____
4	CDPOS	<p>As above (1) (OPTIONAL: Enter filter parameters, e.g. Table Name=LFBK, Field Name=KEY, Change ID=I).</p> <p>Change Width of Output List to 400 Enter Filename: CDPOS.TXT</p>	Change Document Line Item	..	_____
5	LFA1	<p>As above (step 1)</p> <p>Enter Filename: LFA1.TXT</p>	Vendor General Data	..	_____

Appendix 9: Feedback on prototype

Feedback letter 1 – from BDO



Kishore Singh
Doctoral Candidate
University of Southern Queensland
Toowoomba Queensland

Dear Kishore,

COMMENT ON DOCTORAL RESEARCH

Thank you for demonstrating the prototype software developed for your doctoral research entitled " A Conceptual Model for Proactive Fraud Detection In Enterprise Systems: Exploiting SAP Audit Trails to Detect Asset Misappropriation"

You have outlined that the objective of this research is to:

..... determine the feasibility of using technology to automate fraud detection in enterprise systems. Large scale enterprise systems provide the necessary infrastructure for ongoing use of continuous monitoring applications. These applications enable analysis of transaction data in a real- or near real-time basis against a set of predetermined rules. The prototype software has been developed with the intention of assisting an auditor in detecting potentially fraudulent activities in accounts payable. The software takes audit trail data from a SAP system and analyses it for anomalous activities associated with potential vendor fraud. A series of reports and visualisations are produced to support the audit function.

As requested, I have provided comments on the areas you raised:

1. The importance of such a project for auditing in an organisation.

A project of this nature is considered to be of high importance to organisations. It provides a mechanism to proactively monitor fraud risk, a key risk in any organisation. It also demonstrates a commitment to compliance with Corporate Governance Principles and Recommendations as outlined by ASX Corporate Governance Council. In the 2nd Edition of these guidelines, Recommendation 7.2 states:

"The board should require management to design and implement the risk management and internal control system to manage the company's material business risks and report to it on whether those risks are being managed effectively.

The board should disclose that management has reported to it as to the effectiveness of the company's management of its material business risks."

2. The role that automated fraud detection software could play as an auditing tool for

internal auditors. Automated fraud detection software can provide internal auditors with a tool to efficiently assess the presence of fraud within an organisation. This may also be applied to testing the effectiveness of the controls that management may have in place. A tool of this nature can ensure that the management of the risk of fraud can be undertaken on a more regular or continual basis.

3. The desirability of a retrospective analysis software tool implemented on a standalone computer system as compared with a system embedded within an enterprise system.

The benefits of a tool that is outside of the enterprise system is that it can be an independent check of the effectiveness of the controls in place within the enterprise

system. The desirability will relate to the cost, risk and value propositions to the organisation. This will in part be determined by the risk appetite of the organisation, the potential for fraud and the effectiveness of the internal control environment.

4. The functionality of the prototype software, in particular the user interface, reporting and graphical features.

In general, I found the functionality of the tool to be useful. The user interface would require a minimal level of training and some level of understanding of the SAP application, which is a reasonable constraint. The graphs and visualisations clearly communicated a message for the reader. The speed of running the queries was impressive."

5. Any further comments or suggested improvements to the prototype.

There may be potential to automate some of the scripts and perhaps include additional data sets (in addition to AP) to enhance the value of the software. Regression analysis may be a useful feature along with Benford's law to highlight anomalies or unusual patterns.

The above comments are understood to be included in your thesis as an Appendix. These should be taken as professional observations and not an endorsement by BDO.

Thank you allowing me this opportunity to comment on your thesis.

Yours sincerely

John Halliday
Executive Director Advisory
BDO

Feedback letter 2 – from case study company

Mr. Kishore Singh
Doctoral Candidate
University of Southern Queensland
Toowoomba
Australia

Dear Kishore

RE: SAP VENDOR FRAUD DETECTION PROJECT

We are delighted with your prototype software developed with the objective of detecting potentially fraudulent activities in accounts payable. As one of [REDACTED] largest companies who is operating in SAP environment, it is extremely vital to have system based controls in detecting and preventing fraudulent activities. Considering the number of transactions that take place every day, it has become impractical to check each transaction in-detail manually unless they are covered by way of controls in place. In such an environment this software will immensely help to our internal auditors to carry out various tests in detecting frauds and errors.

It is an advantage that we can operate this software on a standalone computer system rather than embedded in our main SAP system because it minimises the disruptions to routine operations and allow retrieving reports at any given time even the on-line system is not available. The dashboard which indicates summary of all reports is a very helpful feature in this software.

In our opinion the functionality of the prototype software should be further extended to other areas in the FICO module such as accounts receivable, fixed assets, general ledger etc., in the near future.

We are happy to work with you on this project. Your recent findings based on testing your software on our data, and highlighted in your report, has helped us to streamline our processes in more meaningful manner.

We wish you all the very best in your future endeavours.

Yours sincerely,

K.M

Manager Finance (Internal Audit)



Feedback letter 3 – from case study company

(Email received from N.J., Financial Director)

Hi Kishore

Thank you very much for your work with us, it was a new perspective to the risks that we carry and methods of identifying some of those using your solution. The learning were very important and valid.

We hope to continue working on the areas we have identified with you and hope for your continued assistance in this matter.

Regards

N.J.

