

***SEMI-AUTOMATIC MATCHING OF
SEMI-STRUCTURED DATA UPDATES***

By:

Gareth William Forshaw B.Sc. (Hons) UCT

Supervised By:

Associate Professor Sonia Berman

A dissertation presented to:

The Department of Computer Science

University of Cape Town

Submitted in partial fulfilment of the requirements for the degree of M.Sc. (Information
Technology)

February 2014



The copyright of this thesis vests in the author. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.

Published by the University of Cape Town (UCT) in terms of the non-exclusive license granted to UCT by the author.

Declaration

I declare that this dissertation is my own work and where appropriate I have acknowledged the work of others. The dissertation is being submitted in partial fulfilment of the requirements of the department of Computer Science for the Degree of Master of Science at the University of Cape Town. It has not been previously submitted for any degree or examination at UCT or any other university.

.....

Gareth William Forshaw

17th day of February 2014

ABSTRACT

Data matching, also referred to as data linkage or field matching, is a technique used to combine multiple data sources into one data set. Data matching is used for data integration in a number of sectors and industries; from politics and health care to scientific applications. The motivation for this study was the observation of the day-to-day struggles of a large non-governmental organisation (NGO) in managing their membership database. With a membership base of close to 2.4 million, the challenges they face with regard to the capturing and processing of the semi-structured membership updates are monumental. Updates arrive from the field in a multitude of formats, often incomplete and unstructured, and expert knowledge is geographically localised. These issues are compounded by an extremely complex organisational hierarchy and a general lack of data validation processes.

An online system was proposed for pre-processing input and then matching it against the membership database. Termed the Data Pre-Processing and Matching System (DPPMS), it allows for single or bulk updates. Based on the success of the DPPMS with the NGO's membership database, it was subsequently used for pre-processing and data matching of semi-structured patient and financial customer data. Using the semi-automated DPPMS rather than a clerical data matching system, true positive matches increased by 21% while false negative matches decreased by 20%. The Recall, Precision and F-Measure values all improved and the risk of false positives diminished. The DPPMS was unable to match approximately 8% of provided records; this was largely due to human error during initial data capture. While the DPPMS greatly diminished the reliance on experts, their role remained pivotal during the final stage of the process.

ACKNOWLEDGMENTS

This research would not have been possible if it was not for the numerous individuals and organisations that provided support and assistance during this study.

I wish to give special thanks to my supervisor, Associate Professor Sonia Berman, her direction and insight during this research, and throughout the MIT program, was invaluable.

Thanks are also due to my employer for their continued patience as well as for the financial support of my academic pursuits.

Lastly, to all my family and friends, your support of my studies in general has been a great source of inspiration and encouragement.

Above all, get obsessed about data

Jerry Yang, founder of Yahoo

TABLE OF CONTENTS

DECLARATION	II
ABSTRACT	III
ACKNOWLEDGMENTS.....	IV
TABLE OF CONTENTS	V
LIST OF FIGURES.....	IX
LIST OF TABLES.....	X
ABBREVIATIONS.....	XI
CHAPTER 1. INTRODUCTION	1
1.1 Introduction to the NGO data set	2
1.2 Objective.....	3
1.3 Research methodology.....	3
1.4 Scope	4
1.5 Data sterilisation	4
1.6 Organization of this dissertation	4
1.7 Conclusion	5
CHAPTER 2. BACKGROUND	6
2.1 Introduction.....	6
2.2 Quality	6
2.2.1 Data quality	6
2.2.2 System quality	9
2.3 Semi-structured data	10
2.4 Data pre-processing	11
2.5 Data matching	13
2.5.1 Data matching methods	13
2.5.2 Data matching evaluation.....	16
2.6 Conclusion	20

CHAPTER 3. CURRENT APPROACH.....	21
3.1 Introduction.....	21
3.2 NGO data set overview	21
3.3 Overview of current data	23
3.3.1 Scale	23
3.3.2 Data storage.....	23
3.4 Current update process	23
3.4.1 Annual and biennial reviews	24
3.4.2 Ad-hoc updates.....	24
3.4.3 Employee updates.....	25
3.4.4 Data quality concerns	25
3.5 Suitability for data matching	26
3.6 Conclusion	26
CHAPTER 4. DPPMS DESIGN	27
4.1 Introduction.....	27
4.2 Design overview.....	27
4.3 Process overview	28
4.4 Component overview	30
4.4.1 Interface.....	30
4.4.2 Database	30
4.4.3 Data Cleaning.....	31
4.4.4 Data matching	31
4.4.5 Rules.....	33
4.4.6 Reporting.....	36
4.5 Evaluation goals	37
4.5.1 Rule validation	37
4.5.2 Record sub-sampling	37
4.5.3 System level testing.....	38
4.6 Conclusion	38
CHAPTER 5. DPPMS IMPLEMENTATION	39

5.1	Introduction.....	39
5.2	Software applications.....	39
5.2.1	MySQL.....	39
5.2.2	Java Code	40
5.3	Interface.....	42
5.4	Reporting and Auditing.....	46
5.5	Detailed system implementation	46
5.5.1	Tokenization.....	46
5.5.2	Data cleaning.....	47
5.5.3	Field-level matching.....	47
5.5.4	Full string verification	48
5.5.5	Available update actions	53
5.6	Conclusion	55
CHAPTER 6. APPLICABILITY TO ALTERNATE DATA SETS.....		56
6.1	Overview of the data sets.....	56
6.1.1	Introduction to data in the health care sector.....	56
6.1.2	Introduction to data in the Financial Services sector.....	58
6.1.3	Similarities to the NGO data set.....	60
6.2	Overview of requirements	62
6.2.1	Precision and Recall	62
6.2.2	Handling of incomplete data	62
6.2.3	Potential security implications	62
6.3	Required system updates.....	62
6.3.1	Rule updates	63
6.3.2	Dictionary lists	63
6.3.3	Interface updates.....	63
6.4	Conclusion	64
CHAPTER 7. TESTING AND EVALUATION.....		66
7.1	Introduction.....	66
7.2	Rule validation	66

7.3	System-level testing	68
7.3.1	Health sector.....	69
7.3.2	Financial sector	70
7.3.3	NGO domain	72
7.4	Domain result comparison	74
7.5	Discussion of results	75
7.6	Observations	76
7.6.1	Damerau–Levenshtein distance metric.....	76
7.6.2	Relocation functionality	77
7.7	Conclusion	77
CHAPTER 8. CONCLUSION		78
8.1	Introduction	78
8.2	Summary	78
8.3	Future Work	79
8.3.1	Quantifying the Recall and Precision compromise	80
8.3.2	Data-set level matching	80
8.3.3	String distance metrics	80
8.3.4	User interface design.....	81
8.3.5	Data mining and data warehousing	81
8.4	Conclusion	81
REFERENCES		82
APPENDIX		89
NGO database schema		89

LIST OF FIGURES

<i>Number</i>	<i>Page</i>
FIGURE 1. AN OVERVIEW OF THE GENERIC DATA MATCHING PROCESS [47].	15
FIGURE 2. DIAGRAM SHOWING AN EXAMPLE NGO HIERARCHY; INCLUDING POTENTIAL PARENT NODES FOR EMPLOYEES, VOLUNTEERS AND ALL MEMBERS.	22
FIGURE 3. HIGH LEVEL PROCESS OVERVIEW.	29
FIGURE 4. DATA MATCHING STEPS.	33
FIGURE 5. THE DPPMS <i>AUTOMATIC - SINGLE ROW ENTRY</i> SCREEN – INITIAL USER INPUT.	44
FIGURE 6. THE DPPMS <i>AUTOMATIC - SINGLE ROW ENTRY</i> SCREEN – FINAL RESPONSE.	44
FIGURE 7. THE <i>VIEW DATA</i> SCREEN – SHOWING THE STAFF BY HOME ENTITY DATATABLE PANEL.	45
FIGURE 8. PATH RESOLUTION FOR AN EMPLOYEE.	50
FIGURE 9. EVENT PATH RESOLUTION FOR A VOLUNTEER (VOL).	52
FIGURE 10. FINAL UPDATE LOGIC.	54
FIGURE 11. OVERVIEW OF THE RULE VALIDATION PROCESS.	67
FIGURE 12. COMPARISON OF RESULT DATA ACROSS THE THREE DOMAINS.	74
FIGURE 13. COMPARISON OF THE RECALL, PRECISION AND F-MEASURE VALUES.	74
FIGURE 14. POTENTIAL HIGH-LEVEL MATCHING FLOW.	80
FIGURE 15. NGO MEMBERSHIP DATABASE SCHEMA.	89

LIST OF TABLES

<i>Number</i>	<i>Page</i>
TABLE 1. DATA QUALITY DIMENSIONS	7
TABLE 2. COMPARISON OF CHARACTERISTICS OF THE THREE KEY DATA TYPES [36].	10
TABLE 3. ERROR MATRIX DEMONSTRATING THE POTENTIAL OUTCOME OF A DATA MATCHING EXERCISE [47].....	18
TABLE 4. SIMILARITIES ACROSS NGO, PATIENT AND FINANCIAL DATA.	61
TABLE 5. EXTRACT FROM THE SCENARIO PERMUTATION MATRIX OF THE NGO DATA SET.	68
TABLE 6. SUMMARY OF THE HEALTH SECTOR TEST RESULTS.	69
TABLE 7. SUMMARY OF THE RECALL, PRECISION AND F-MEASURE VALUES FOR THE HEALTH SECTOR.	70
TABLE 8. SUMMARY OF THE FINANCIAL SECTOR TEST RESULTS.....	71
TABLE 9. SUMMARY OF THE RECALL, PRECISION AND F-MEASURE VALUES FOR THE FINANCE SECTOR.	71
TABLE 10. SUMMARY OF THE NGO TEST RESULTS.....	72
TABLE 11. SUMMARY OF RECALL, PRECISION AND F-MEASURE FOR THE NGO MEMBERSHIP DOMAIN.	73

ABBREVIATIONS

- CSV - Comma-separated values file
- DPPMS - Data Pre-Processing and Matching System
- FN - False negative
- FP - False positive
- GIS - Geographical information system
- HR - Hit Rate
- HR - Human resources
- IT - Information technology
- JDBC - Java Database Connectivity
- JSF - JavaServer Faces
- KYC - Know your customer
- MNO - Mobile network operator
- MS - Microsoft
- NC - National Council
- OTF- Off-the-shelf
- PC - Personal computer
- PII - Personal identifiable information
- PPV - Positive Predictive Value
- RDBMS - Relational Database Management System
- SA - Republic of South Africa
- SAHIA - The South African Health Informatics Association
- SRM - Structured relevance model
- SSL - Secure Sockets Layer
- TN - True negative
- TP - True positive
- TPR - True positive rate
- UI - User interface
- US - United States of America
- XML - Extensible Markup Language

Keywords: Data matching, semi-structured data, schema matching, data prep-processing, relational databases, database schemas.

CHAPTER 1. INTRODUCTION

As McKendrick states in *Today's Data Systems Not Quite Ready for Real-time*, the reality of modern day business is that organizations need to deliver the right information, at the right time [1]. Regardless of the sector or industry, the importance of maintaining accurate and well-structured data sets is indisputable. For the financial industry, this need is often driven by governance and legislation [2]; in the health sector, this need is often driven by confidentiality concerns [3] and the wellbeing of the patients [4].

This study was driven by the experiences of a large, global, NGO which was struggling to accurately maintain its Volunteer and Employee database due to an ad-hoc and unstructured update process. This particular NGO has extremely complex internal organisational hierarchies and sub-structures which further exacerbated their struggles with data management. The above factors led to their database being onerous and cumbersome to maintain, highly error prone, and most importantly, viewed as largely untrustworthy by the users. Initial manual attempts to validate and verify the NGO's current data set have largely failed due to the sheer complexity of its hierarchies and structures. Human-led data validation has proven to be overwhelming as people's knowledge of a data set is constrained by their geographic location and their level within the organisation (i.e. individuals lower in the hierarchy have limited upwards knowledge and individuals higher up the hierarchy have a limited downwards view). Initial attempts included using the postal service and email to manually validate member and entity data. These efforts were, however, fraught with risk and bore limited success.

As the initial investigation progressed, two key themes became apparent. The first was that the challenges they were facing were not limited to just NGOs. A number of industry sectors were experiencing similar issues with poor quality semi-structured data sets. Secondly, it became clear that while the initial data capturing phase was flawed and needed improvement [5], the immediate requirement was for an improved data pre-processing and matching approach to data updates. These discoveries spurred the development of the Data Pre-Processing and Matching System presented in this dissertation, hereafter referred to as the DPPMS. While initially developed for the NGO, it was subsequently utilised in the health care and finance domains as well, in order to ascertain if the approach was widely applicable.

1.1 Introduction to the NGO data set

Large enterprises are able to deploy and maintain mature and extensive database applications. These enterprise-grade database systems include extensive functionality that guarantees privacy, data integrity, robustness, security and high availability. These systems are nearly always catered to by a large and often dedicated staff contingent. They are also well covered by relevant policies and procedures. While all of the above ensure that enterprise database systems enjoy near faultless operation and 99.9999% uptime [6; 7], the costs of deploying and maintaining such systems are prohibitively high.

Non-governmental organizations (NGOs) such as charities and community based organizations do not have access to the funds or infrastructure required to deploy such stable and effective database application systems [8]. However, the reality is that organizations often have very similar database needs to large enterprises; this includes needing access to reliable, secure database systems capable of handling enterprise-sized data processing requirements.

The majority of NGOs are not capable of planning, deploying or maintaining enterprise-level database systems. Barriers to them entering this space include not having the financial means, lack of suitably skilled staff, lack of focus and understanding from senior management as well as complex system requirements not adequately catered for by off-the-shelf (OTF) database management systems. NGOs often have extremely complex structures, the complexity of which can sometimes even eclipse that of their counterparts in the private sector. While the complexities experienced by NGOs are many, the key issues they face include:

- Complex organisational hierarchies and sub-structures.
- People can operate simultaneously in multiple roles at multiple levels of the hierarchy.
- Most NGOs are largely volunteer-driven organisations, so updates to databases are obtained from a multitude of sources including, emails, spreadsheets, Word documents, phone messages and hand written letters. It can thus be a complex exercise to import data from such a multitude of semi-structured data sources.
- Vast geographical distribution of localised expert knowledge.
- Multiple contributors with limited visibility and collaboration.

- Lack of clear processes and guidelines.
- Lack of data validation and verification mechanisms.

The above factors lead to NGOs having databases that are problematic to maintain, error-prone, and unreliable.

1.2 Objective

In all three sectors observed - NGO, financial and health care - the import of semi-structured data into existing, structured, databases is often done manually. Manual matching is the term used to refer to the clerical approach to data matching, and is highly inefficient: local experts with knowledge of the data manually match new information to existing data before executing required updates. Even with dedicated knowledge experts, the error and failure rates of this clerical data matching process remain unacceptably high. To mitigate this, this dissertation investigates the development of a data pre-processing and matching system that would be able to successfully match input fields with existing database fields, identify missing or incomplete data, infer missing field data and then execute any relevant database updates. An online system was developed that enables users to upload single data records, upload batch updates in CSV files, view processed and existing data, and confirm or reject uncertain updates where system confidence is low. The DPPMS approach was then evaluated to measure the benefit compared to using existing manual methods.

1.3 Research methodology

A Quantitative Experimental approach served as the primary research methodology. As part of this approach, two valid processes and a third experimental process were applied to identical data-sets. A Quantitative Evaluation based methodology was then used to assess each of the three result-sets in terms of the evaluation goals as specified in Chapter 4.

1.4 Scope

The DPPMS, as implemented and described here, consists of the mechanisms required to upload, pre-process, data match and then record the result for each record provided. The DPPMS purposely does not contain field specific matching functionality, such as expanded address matching algorithms, as these concepts are adequately covered by existing research material and methodologies [9; 10]. The DPPMS was also not developed to cater for all of the realities a production system would face such as scalability, security and interface usability [6; 11]. The DPPMS was developed to allow for the updating and relocation of entities within the data set. Changes such as the addition of new members, deletion of current members and the renaming of members were outside the scope of this research and were not implemented.

1.5 Data sterilisation

Due to the fact that the original data sets contained personally identifiable information, certain fields were removed for the purpose of this study. Where appropriate, all such values were replaced by suitably synthesised data that contained flaws and inaccuracies similar to what could be expected in a real world data set. The data used for this research contained no personally identifiable information and the results could not be used to infer any form of identification.

1.6 Organization of this dissertation

This dissertation has the following chapters:

- Chapter 1: This chapter, an introduction
- Chapter 2: A literature review
- Chapter 3: Overview of the current processes in place
- Chapter 4: Overview of the DPPMS and its design
- Chapter 5: Implementation of the DPPMS
- Chapter 6: Applicability to alternate data sets
- Chapter 7: Testing and evaluation of the DPPMS

- Chapter 8: Conclusion

1.7 Conclusion

This chapter has defined the problem addressed in this thesis and outlined the solution. In summary, this work aims to develop a system to semi-automatically pre-process and match semi-structured data to data in an existing database, and to evaluate its effectiveness for accurately capturing database updates.

The following chapter consists of a literature review of relevant literature.

CHAPTER 2. BACKGROUND

2.1 Introduction

As Mansuri quite aptly wrote, “Database systems are islands of structure in a sea of unstructured data sources” [12]. A system that can successfully pre-process and data match semi-structured data covers many fields within computer science. The topics of data quality, semi-structured data, data matching and data-preparation are explored below.

2.2 Quality

The overall concept of quality formed an integral part of this dissertation. The two most significant aspects of quality are overall system quality and data quality.

2.2.1 Data quality

In today’s competitive world of knowledge-driven decision making, poor data quality is one of the single greatest threats facing organisations. Poor quality data, often referred to as bad data, is defined as data that is incorrect or incomplete [13; 14]. The most common manifestations of bad data include missing and incorrect attribute values as well as duplicate occurrences of the same data. Good quality data, as defined by Juran, is data that is "fit for their intended uses in operations, decision making and planning" [15].

Another core concept related to data quality is data reliability. Agmon proposed three states of data reliability: internal, relative and absolute reliability. These three states represent the data’s compliance with commonly accepted norms, fulfilment of user requirements and the similarity of the data to real world entities [16]. In the context of data matching and this dissertation, the concept of absolute reliability is core as all input data represents an individual and her position in a well-defined organisational hierarchy.

When reviewed, the available data quality statistics are nothing short of alarming. In the United States of America (US), it is estimated that 60% to 90% of all operational data is bad data [17]. It is further estimated that bad data costs US businesses over US\$ 600 billion a year in loss and damages [18]. The risks associated with bad data can sometimes be difficult to quantify. While cost and lost man hours can be measured relatively easily,

upset customers, delayed decision making and lost credibility are all further potential consequences of bad data [19]. The impact of those consequences can be long lasting and extremely difficult to mitigate.

While the impact and cost of bad data can be difficult to accurately estimate and comprehend, agreeing on a set of data quality dimensions is equally challenging. Attempts to structure data quality into defined dimensions have largely failed, with an almost infinite number of dimensions currently proposed. The dimensions that Wang proposed in 1993 are still the most accepted. Table 1 summarises his sixteen proposed data quality dimensions [13; 20].

Dimension	Description
Accessibility	Data must be available or easily and quickly retrievable.
Accuracy	Data must be correct, reliable, and certified free of error.
Appropriate Amount of Data	The quantity or volume of available data must be appropriate.
Believability	Data must be accepted or regarded as true, real, and credible.
Completeness	Data must be of sufficient breadth, depth, and scope for the task at hand.
Concise Representation	Data must be compactly represented without being overwhelming.
Ease of Understanding	Data must be without ambiguity, and easily comprehended.
Ease of Manipulation	Data must be easy to manipulate and re-task.
Interpretability	Data must be in appropriate language and units, and the data definitions must be clear.
Objectivity	Data must be unbiased (unprejudiced) and impartial.
Relevancy	Data must be applicable and helpful for the task at hand.
Representational Consistency	Data must always be presented in the same format and compatible with previous data.
Reputation	Data must be trusted or highly regarded in terms of their source or content.
Security	Access to data must be restricted, and hence, kept secure.
Timeliness	The age of the data must be appropriate for the task at hand.
Value-Added	Data must be beneficial and their use advantageous.

Table 1. Data quality dimensions

While all sixteen dimensions should ultimately be used to define “good data”, certain dimensions are more critical than others.

The majority of data consumers have three core priorities; they want data that is easy to consume, relevant and most importantly, accurate [19]. During the development and assessment of the DPPMS, the four dimensions identified as being the most critical included believability, completeness, consistent representation and accuracy.

When reviewing the various data quality dimensions outlined above, it becomes apparent that data quality is a multi-faceted challenge that is heavily influenced by the intended use of the data [21]. A holistic and structured approach needs to be taken when assessing data quality issues and the quality of data cannot be enhanced in isolation of the original data source [18]. It can be concluded that the only way to guarantee holistic data quality is to improve both the acquisition and processing systems. This conclusion was particularly motivating for this work as it validates the approach taken to developing the DPPMS.

2.2.1.1 Data quality during data collection

As concluded above, it is imperative that data quality intervention efforts are focussed on the data source rather than attempting to rectify data post-collection. The emphasis should always be on collecting data correctly the first time [22]. The challenge is pronounced in scenarios where the original data source may no longer be available or may not even be known [23]. Names, be they for people or places, are often the most error prone fields [24]. Studies have shown that up to 20% of surnames and 25% of first names are captured inaccurately.

The data collection process can be erroneously blamed for data quality issues, as incorrectly captured and missing data are only two of the multitudes of data quality dimensions [18].

The key to holistic data quality management is to ensure that the end-to-end data collection, management and retrieval process is well defined and documented [25]. While Skoogh proposes the use of an exceedingly formal framework to achieve this, such a formal approach will not always be practical or achievable [26]. The important point is, by effectively managing the entire data lifecycle, it is possible to sustain a high level of data quality.

2.2.1.2 Data cleaning to improve data quality

Data cleaning, also referred to as data cleansing, is the process of identifying and then taking action to remove errors from within a data set [27]. Data cleaning, by its very nature, is an inefficient process with an estimated 80% of resources being spent on the error identification phase with only 20% spent on mitigating and rectifying the bad data.

The most commonly resolved issues during data cleaning include the removal of invalid characters, invalid options and out-of-range values; interrogation of outlier values; resolution of missing values and validation of ranges [28; 29]. Advances in hybrid cleaning algorithms allow for advanced statistical approaches to be combined with existing conventional constraint checking methods [30]. These hybrid approaches improve error rates compared to traditional rule-driven counterparts. Due to the nature of the data used for this dissertation, a traditional approach was taken to data cleaning and constraint checking.

Data cleaning can be a misleading concept as all too often it is seen as a once-off intervention. Data quality management needs to be on-going; data cleaning is by no means a standalone solution for a data set's poor quality [18].

2.2.2 System quality

The two corner stones of total system quality are verification and validation, both of which add credibility and validity to the DPPMS [31]. Verification ensures that the DPPMSs overall structure is correct while validation ensures that the DPPMS is able to produce results and form conclusions that are accurate. Verification and validation are inter-dependent; an issue with one will often undermine the quality of the entire system. During the development of an expert system, researchers are often more concerned with the verification and validation phases while stakeholders are more concerned with the final assessment and evaluation phases [32]. While this dissertation attempted to take a balanced approach to overall system quality; validation and credibility received the most attention as they were deemed to be the least dependent on the underlying implementation layers and more indicative of the actual quality of the system.

2.3 Semi-structured data

Semi-structured data is seen by many as one of the greatest challenges currently facing data researchers [33]. Firstly, methods and tools that have proven successful with transforming and leveraging structured data are ineffective when applied to semi-structured data. Secondly, most data on the Internet is in the form of semi-structured data, for example, XML files and email. As the amount of semi-structured data on the Internet increases, our inability to mine it effectively hampers our ability to effectively engage with this vast knowledge resource. Table 2 compares structured, semi-structured and unstructured data [34].

Unstructured	Semi-structured	Structured
<ul style="list-style-type: none"> • Data can be of any type • No set format or sequence • No formal rules • Not predictable 	<ul style="list-style-type: none"> • Attempt to reconcile database and document "worlds" • Organised in semantic entities • Similar entities are grouped together • Entities in same group may not have same attributes • Order of attributes not necessarily important • Not all attributes may be required • Size of same attributes in a group may differ • Type of same attributes in a group may differ 	<ul style="list-style-type: none"> • Data is organised in semantic chunks • Similar entities are grouped together • Entities in the same group have the same descriptions • Descriptions for all entities in a group: <ul style="list-style-type: none"> ○ have the same defined format ○ have a predefined length ○ are all present ○ and follow the same order

Table 2. Comparison of characteristics of the three key data types [36].

As can be observed from Table 2, semi-structured data does not conform to a formal structure or data model. The key challenges of semi-structured data are that the schema can be constantly evolving, not given in advance, not be implicit, provided partially or descriptive as opposed to prescriptive.

Types within semi-structured data also present challenges as objects and attributes are not strongly typed and objects in the same collection can have different representations [35]. A further compounding factor with semi-structured data is its relative incompatibility with traditional relational queries [36]. This arises from the issue that by their very nature, relational database queries often assume that all records within the database are complete but semi-structured data rarely complies with this norm. Due to these characteristics, semi-structured data is often said to be self-describing as it does have relevant markers and semantic elements but it lacks structure and an explicit schema. While the input data described throughout this research was more representative of bad data rather than semi-structured data, the input strings took the form of semi-structured data: tokens were often clearly separated, but their number, order and types varied.

2.4 Data pre-processing

Pyle [39] describes the primary purpose of data preparation as creating a prepared data set that is of maximum use to its owner. She emphasises that data preparation should disturb the layout of the data as little as possible, as data preparation is not about making unnecessary changes to data, but about maximizing the potential of the data set to yield positive results. This is best summarised by Ronald Coase, "If you torture the data long enough, it will confess" [37].

The three disciplines that currently have the largest vested interest in data pre-processing include data mining, data matching and statistical surveys [38]. It is estimated that these sectors dedicate upwards of 40% of their database related resources to data input and pre-processing tasks [39]. In the data-mining sector specifically, it is estimated that the data cleaning and pre-processing tasks can actually take up to 80% of the resources [40].

In these three fields, input data should ideally have the following structure [41]:

- Data should be in a row and column, table format.
- Rows should contain information on individual entities.
- Columns should contain attributes, or fields, of a defined type.
- The attributes can either be continuous or categorical.

In the context of data matching, data pre-processing is the initial and most critical step.

The effective cleaning and standardisation of the input data increases the positive data match rate exponentially [42]. The primary reason for this is that effective data pre-processing steps aggregate and transform what are often a multitude of chaotic data sources into a single consistent and homogenous data set [43; 44]. It must be noted that these transformation steps are especially critical when dealing with name and address fields where the potential for misinterpretation is so great [9; 10; 45].

As discussed, the three main quality concerns are partial records and data that is inconsistent and noisy [29]. Missing data values can be handled by either ignoring the record, determining a likely value, entering missing data manually, or by replacing blank values with a constant such as “NA” or an average [46]. Inconsistent and noisy data is often approached using regression, binning or clustering [29].

To ensure a consistent and accurate approach to data cleaning, it is best to confront data cleaning as a unified process. Developed specifically for data matching, Christen proposes a data pre-processing process that includes the following steps [47]:

- Removal of unwanted and unnecessary characters and words.
- Correcting spelling, acronyms or other slang and colloquial terminology.
- Segmentation of attributes into well-defined and consistent attributes.
- Verification of attribute values.

In the DPPMS, the initial data pre-processing step includes the replacement of all delimiting characters with a standard delimiter. All other positively matched stop words and characters are then removed [48]. After data normalization and smoothing, a dictionary based confirmation process is used to identify and correct spelling and terminology issues. By using a dictionary centred approach, verified domain knowledge can be used to effectively validate and clean the data [49]. Where missing attributes were identified, attempts were made to infer and then populate the missing attributes using a Structured Relevance Model (SRM) approach. SRMs are temporary models that are constructed based on the theory that, within semi-structured data, the value of a missing attribute can sometimes be deduced from other populated fields within the provided data [50]. This approach proved particularly effective in cases where the input data was an update and the database already contained attributes relating to the entity being updated.

While data preparation and pre-processing research is growing, it is by no means a mature discipline. The application of data pre-processing methods is not yet routine and does have its difficulties [51]. The future holds some exciting developments, primarily in the fields of intelligent and self-learning data pre-processing methodologies.

2.5 Data matching

Data matching, also referred to as data linkage or field matching, is a technique used i.a., to combine multiple data sets into one [47]. The fields from two or more data sources are matched based on semantic similarities, such as the likeness of terms, and then transformed into a common format to form a single data set [52; 53]. While the motivations for data matching differ, the process and challenges involved are universal. A number of breakthroughs have been made with regard to automatic schema matching as well as the use of hybrid combinations of matching methods [44].

Two of the greatest sources of frustration during the data matching process are poor data quality and poorly structured data [54]. The dangers of poor quality data are all multiplied significantly during data matching exercises. Semi-structured data is a prime example of a data set type that is notoriously difficult to successfully data match as it is not formally arranged in a conventional structure or framework. This is often compounded by the fact that expected fields may be missing or may be present but contain data that is difficult to describe.

The British Electoral Commission experienced a number of these challenges while data matching semi-structured data from 10 separate data sources. They invested over £1,200,000 pounds and achieved a data-match rate of less than 57% [55]. Due to these results they were ordered to appear before a House of Commons sub-Committee [56].

2.5.1 Data matching methods

Numerous techniques can be used for data matching; the most common include [57; 58]:

- *Probabilistic* - all available fields are used in an attempt to locate a match. Fields are not required to match exactly and a weighting system is often used to score potential matches between fields. The score of each potential match is then reviewed and the

relative confidence in each potential match determined. Fields typically used include names and addresses.

- *Deterministic* – Often referred to as exact linkage, this technique relies on there being a unique identifier present that enables an exact match between fields. Fields that are typically used include bank account, social security and tax numbers.
- *Judgmental* – Judgmental matching is used when an exact match cannot be found, but a partial or inferred match can be located. These partial matches can be located using either an automated or manual matching system.

Other available techniques include manual matching, rules-based and fuzzy approaches. These techniques are rarely applied in isolation [47], and most approaches use a common set of descriptors and data characteristics in an attempt to match data, including actual attribute values, element names, constraints, data structure, inter-element relationships and expert knowledge such as synonyms and terminology [59].

The key data matching steps are also largely universal for all the above techniques [47]:

1. Pre-processing of input data.
2. Indexing of the cleaned data.
3. Comparison of records.
4. Classification of results into matches, non-matches and potential matches.
5. Evaluation and presentation of the matches.

The flow of these steps in a generic data matching process is presented in Figure 1.

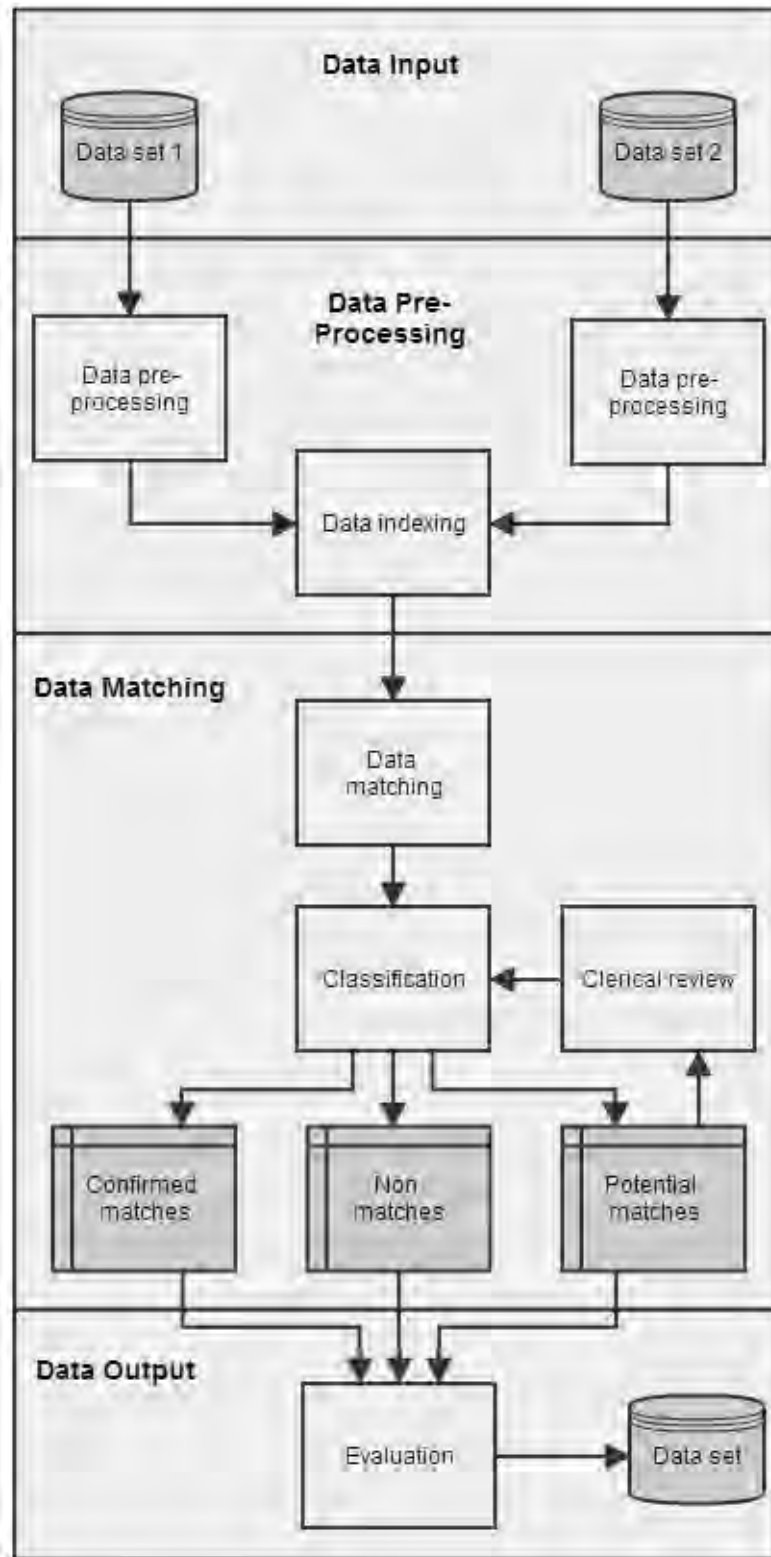


Figure 1. An overview of the generic data matching process [47].

While most matching techniques handle direct matches with relative ease, it is partial and composite matches that often differentiate the effectiveness of a particular method.

It is interesting to note that while a completely automated data matching system would be ideal, the reality is that ongoing interaction from expert users is needed [60]. The benefit of using an automated system is that the required interaction with an expert is the last step in the process and is limited to reviewing of potential matches and any other required interventions. Despite extensive automation efforts and system validation by experts, data matching will continue to be a subjective process that is prone to error [61].

2.5.1.1 Damerau–Levenshtein distance algorithm

While not exclusively related to data-matching, the Damerau–Levenshtein distance algorithm played a key role in the DPPMS and needs to be briefly explored. Named after Frederick J. Damerau and Vladimir I. Levenshtein, the main purpose of this algorithm is to determine the distance, or difference, between two text strings [62]. The algorithm works by comparing two text strings then calculating the minimum number of operations required to transform the first string into the second string; available operations include insertion, deletion, substitution and transposition. The resulting metric is a good indicator of the likelihood that two fields are in fact a match and that any minor discrepancies are likely to be a result of misspellings or other input errors [63].

In the context of this dissertation, the Damerau–Levenshtein algorithm was chosen over the Levenshtein algorithm as the Damerau–Levenshtein algorithm includes transpositions while the Levenshtein algorithm does not. By including transpositions, it has been theorized that the Damerau–Levenshtein algorithm can detect upwards of 80% of all human misspellings within text [64].

2.5.2 Data matching evaluation

Without effective evaluation and measurement techniques, the result of any data matching process would be of limited value [47]. Data match evaluation can be divided into two core concepts, evaluation methods and measurement techniques [29].

2.5.2.1 Evaluation

The primary objective of the evaluation phase is to demonstrate reliability and robustness [65]. For data matching, the majority of evaluation methods include the principle of comparing actual matched results with an oracle, or golden record set, to determine the quality of the match results [47; 66].

The majority of oracle data sets originate from four primary sources [29; 47]:

- *Clerical review of matched data*– A data set created by expert users who manually review and match the data. This is an extremely costly and time consuming process, especially when working with large and complex data sets.
- *Dedicated test data* - A number of dedicated test data sets are available in the public domain for testing purposes. These quasi-real data sets contain realistic distributions that allow for accurate and realistic test results while avoiding the privacy and confidentiality pitfalls of testing with real-world data.
- *Previously generated match data* – In certain scenarios, previous research may have generated an accurate result set that can be reused to validate a new methodology on the same data set.
- *Generation of synthetic test data* – Researchers are able to generate data sets that have certain known characteristics as well as similar attributes to that of real world data sets. By controlling the attributes and characteristics of the input data, the expected results of the data matching process can be modelled.

During evaluation of the DPPMS, both the clerical review and synthetic data approaches were used. Smaller data sets of up to 1000 records that had been clerically reviewed were used during the initial development stage. During final evaluation, data sets of up to 3000 records were used for testing and experimentation.

2.5.2.2 Measurement

Once an oracle data source has been established, expected versus actual data match results can be reported. For the DPPMS, all results, including any error or success messages as well as the affected row ID, were automatically written to a standalone table

in the database, this table was then queried as part of the evaluation process. For easy reference, the result data was also available to view via the Web interface.

As part of the match comparison process, four key metrics are generated [67]:

- *True positives*: Records that have been identified as matches and are actual matches: a successful decision by the data matching system.
- *False positives*: Records that have been identified as matches but are not actual matches: a failure of the data matching system.
- *True negatives*: Records that have not been identified as having a valid match and do not have a valid match: a successful decision by the data matching system.
- *False negatives*: Records that have not been identified as having valid matches but do in fact have a valid match: a failure of the data matching system.

Ideally, the main aim of data matching is to correctly identify as many true matches as possible while ensuring that the number of false positives and false negatives remains as low as possible. In practice though, false negatives and false positives are often inevitable. The relationship between Actual Classifications and Computed Classifications is best illustrated using an error matrix, an example of which is included in Table 3.

		<i>Computed Classification</i>	
		Matches	Non-matches
<i>Actual Classification</i>	Matches	True Positives (True matches)	False Negatives (False non-matches)
	Non-Matches	False Positives (False matches)	True Negatives (True non-matches)

Table 3. Error matrix demonstrating the potential outcome of a data matching exercise [47].

To derive true meaning from the above metrics, further analysis is required. The following measures are commonly used when analysing data match quality, where TP is True Positives, TN is True Negatives, FP is False Positives and FN is False Negatives [47; 48]:

- $$accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

Accuracy is best used for balanced classes where the number of matches and non-matches are similar. Due to the very nature of data matching, accuracy is not deemed a suitable measure for data matching quality.

- $$precision = \frac{TP}{TP + FP}$$

Precision is used to calculate the ratio of true matches to proposed matches. This is a good measure of how precise a particular data matching method is. Precision is sometimes referred to as the Positive Predictive Value (PPV).

- $$recall = \frac{TP}{TP + FN}$$

Recall is used to quantify how many actual, or true, matches have been correctly marked as true matches. Recall is sometimes referred to as the True Positive Rate (TPR) or Hit Rate (HR).

- $$fmeasure = 2 * \left(\frac{precision * recall}{precision + recall} \right)$$

F-Measure is used to determine the harmonic between Recall and Precision. F-Measure is sometimes referred to as the f-score or f₁-score.

- *Specificity*: This is calculated using $specificity = \frac{TN}{TN + FP}$

Specificity is sometimes referred to as True Negative Rate (TNR) but should not be used for data matching evaluation due to the ability of TN to skew the result.

- *False positive rate*: This is calculated using $falsepositiverate = \frac{FP}{TN + FP}$

False positive rate is sometimes referred to as fall-out but should not be used for data matching evaluation due to the ability of TN to skew the result.

Ultimately, finding the ideal data matching result involves a compromise between Precision and Recall [68]. Depending on the type of data being matched, a higher Recall may be desired while with other data sets a higher Precision may be desired. For example, the health and financial sectors often run mission critical systems where

erroneously including an individual in a data set may result in substantial repercussions. Examples of such scenarios are medication being incorrectly prescribed to a patient or the erroneous approval of a loan to a high-risk borrower. High-risk scenarios often require a compromise by targeting a high Precision rate with a low Recall rate. On the other hand, NGOs and marketing companies frequently perform tasks such as the generation of mailing lists. The repercussions for erroneously including an individual in a mailing list are relatively limited; there is in fact more risk in the erroneous exclusion of individuals. Due to the nature of their work, these types of organisations generally settle for a high Recall rate and a lower Precision rate. While all three sectors have multi-faceted data-requirements that can be problematic to generalise, their goal is the same, to generate accurate data sets with minimal risk. As this dissertation is based on a data quality issue, the focus was on improving Precision and F-Measure values; Recall had a lower priority.

2.6 Conclusion

This chapter introduced the concepts of data quality, semi-structured data, data pre-processing and data matching. Within data matching, the available evaluation and measurement techniques were explored. It was stated that the metrics to be used as part of this works evaluation phase were Precision and F-Measure, with Recall being used to a lesser degree.

In conclusion, while there is sufficient literature available, the data-mining sector provided one of the best literature bases. This was due to the emphasis that data mining places on data quality.

The following chapter reviews the current approach being used for data-matching and the challenges being encountered.

CHAPTER 3. CURRENT APPROACH

3.1 Introduction

Prior to development of the DPPMS, a thorough review was done of the current data set, data quality concerns and data management processes. This review focussed primarily on the experiences of the NGO which was struggling to accurately maintain its Volunteer and Employee database due to an ad-hoc and unstructured update process.

3.2 NGO data set overview

The NGO data set represents the personal information as well as the position in the hierarchy of all Employees (paid) and Volunteers (unpaid). The structure of the NGO is complex, consisting of a five-layer hierarchy - Global Council, (GC), National Council (NC), District, Circuit and Church - where any layer can have associated Committees, Activities and Events.

These concepts are defined further below:

- **National Council:** All National Councils are children of the single global council.
- **District:** All Districts are children of a National Council.
- **Circuit:** All Circuits are children of a District.
- **Churches:** All Churches are children of a District.
- **Events:** Can be a child of an Activity, Committee, Church, District, Circuit or National Council. This would be a once off, non-recurring item. Examples of these would be AGM's, conferences and council meetings.
- **Activities:** The lowest level in the hierarchy, this is a reoccurring, structured group that can only be a child of a Church.
- **Committee:** All Committees are children of an Event, Activity, Church, District, Circuit or National Council. They can involve multiple Volunteers as well as Employees. Examples would be organising Committees for conferences as well as structures such as Church councils.

- **Member Type:** All individuals within the data set are either unpaid **Volunteers** or paid **Employees**. Employees are nearly always also members of the NGO in their personal capacity; however, their personal involvement is governed by relevant policies and business rules. Volunteers can be associated to Churches, Activities, Events and Committees. Employees can be associated with Committees, Events, Churches, Districts, Circuits, and National Councils.
- **Role:** There are a fixed number of roles that can be held at any level within each structure. Certain roles such as presidents, secretaries and treasurers, are extremely structured and are carefully managed via succession planning, business rules and pre-requisite experience and qualifications. Others such as stewards and administrative roles are governed by fewer constraints.

There are no limitations on the number of children each parent can have. For example, each Church can have multiple Committees, Activities and Events associated to it, as well as multiple Employees and Volunteers.

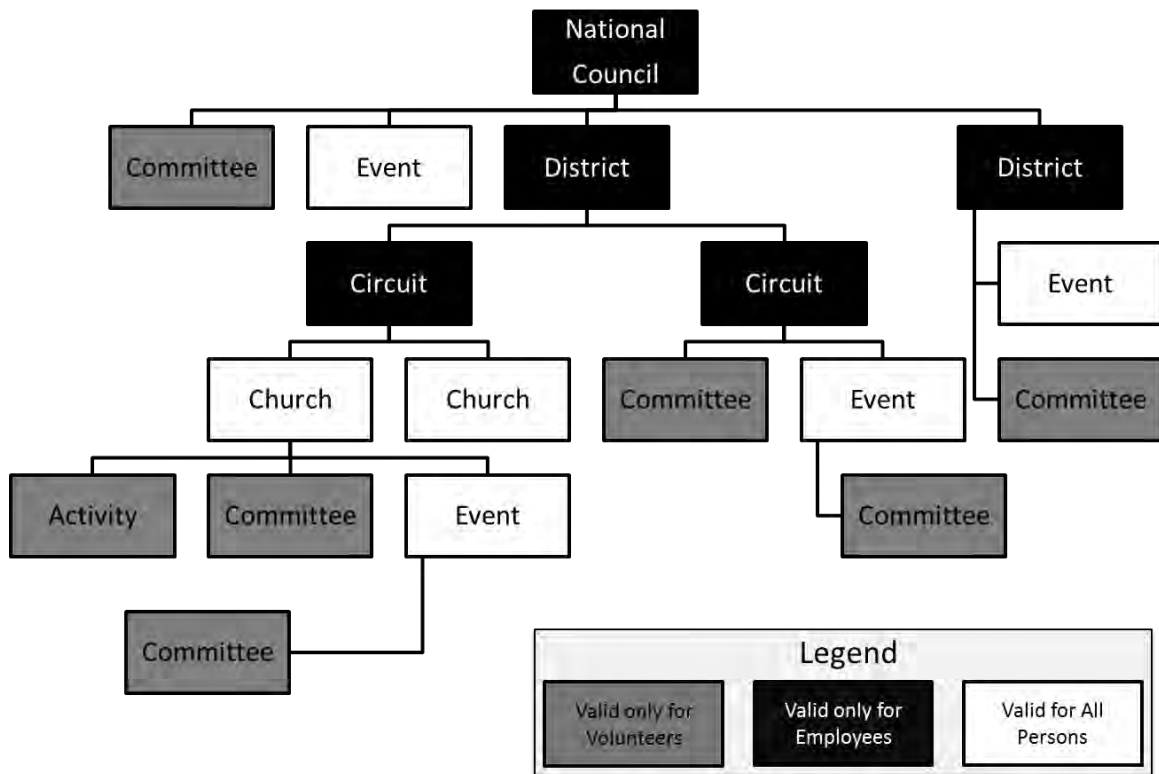


Figure 2. Diagram showing an example NGO hierarchy; including potential parent nodes for Employees, Volunteers and all members.

3.3 Overview of current data

3.3.1 Scale

There are currently 2.4 million Members in 6 Southern African countries. 180 000 of these are Volunteers actively partaking in Activities and Events. There are a further 674 paid Employees. There are 12 Districts, 128 Circuits and 450 Churches.

Each Church on average has 7 on-going Activities at any one time. The number of Events and Committees varies per structure. Churches on average can have approximately 24 Events per year while other structures typically have fewer than 12 a year. Churches on average can have approximately 12 Committees at any one time, while other structures typically have fewer than 4.

3.3.2 Data storage

Currently there is no centralized storage platform for membership details. A central membership registry is maintained; however, this is currently storing limited details for each member and is not accessible by the majority of members. Entities such as Churches, Circuits, Districts, Activities, Events and Committees are responsible for defining their own data storage approaches. The majority use Microsoft (MS) Word, MS Excel, MS Publisher or paper-based filing systems. This was one of the greatest sources of frustration during this research. These frustrations are shared by a number of executive members. Incompatibility and inflexibility was a primary motivator for the DPPMS as a means of importing the above data into a central repository and maintaining it there, in an accurate and efficient manner.

3.4 Current update process

Updates are motivated by deaths, resignations, emigration, marriages, rectification of mistakes, change of addresses, phone numbers and any other such triggers. Updates can also be triggered by members joining, moving between or departing from entities such as Churches, Events, Committees and Activities. Three processes are used to maintain the database: ad-hoc member updates, annual membership reviews and structured Employee data updates.

3.4.1 Annual and biennial reviews

All non-executive positions for Churches, Circuits, Districts, Activities, Events and Committees are elected annually. During these meetings, a directory is published that contains all relevant membership details. Consolidation of the various data sources into this one directory is a laborious manual process.

Members present are expected to review the directory and provide any updates by returning a hand written form to the organizers. Members not present are posted a copy of the directory and can provide updates via email, postal service or via the phone. The process is owned by the secretary of the relevant entity and overseen by the president.

Secretaries can be seen as expert users as they often have an extremely good working knowledge of their entity and the individuals involved. Some concerns observed with this process are:

- A large number of updates are generated within a very short time period.
- Despite the use of forms, the quality and structure of data collected is poor.
- The review process occurs in isolation from the rest of the organisation.
- The format of the published directories varies between different entities.
- The technical skill and knowledge levels of members can vary greatly.

Executive members within Churches, Circuits, Districts, Activities and Committees are elected biennially, are governed by the NGO's Doctrinal Standards and General Rule publications, and require approval from the relevant governing structures. Executive membership details are self-verified upon election and are generally of a good quality.

Overall, the annual and biennial data reviews generated a large number of updates that were relatively straight-forward to action. When reviewing these update sets from 2011, 2012 and 2013; approximately 58% of all updates were processed immediately without any errors, 22% were successfully applied after the intervention of an expert user, 12% were returned to the member for clarification and 8% did not require an update.

3.4.2 Ad-hoc updates

Perhaps the greatest source of frustration and inaccuracy with the membership database came from the ad-hoc update process. Throughout the year, secretaries are responsible for

processing any membership updates that they receive in a multitude of formats, most commonly via a phone call, letter, email or word of mouth. These updates often take the form of a string, for example: Gareth Forshaw, St. Julian's Church, Cape Town, WA1 Committee - president

The quality of some of these updates was poor. They would often exclude key criteria and context data such as names, areas and parent entities. Some updates observed were nothing more than a handwritten note consisting of a first name and telephone number.

On average, 62% of the ad-hoc updates were completed successfully using the secretary's expert knowledge; the remaining 38% required further clarification from an expert user or the user initiating the update.

3.4.3 Employee updates

Updating of all permanent Employee data was handled out of a centralized office. This data was generally of a good quality in line with industry standard Human Resources best practices. The only identified risk was that it was not being fed downwards through the organisation very effectively. The DPPMS was developed to handle both Employee and Volunteer data.

3.4.4 Data quality concerns

Both prior to and during this research, various executive members voiced their concern over the poor data processes and systems. Risks that they routinely identified included:

- The amount of wasted effort and general overhead generated by the efforts to mitigate the bad data was crippling.
- Succession planning for the executives was limited to individuals who know the data set, the members, as well as the organizational structure. In certain areas, the membership data was in such bad shape that exemptions had to be granted to extend secretarial terms. Handing over the data was impractical to the point that if the current secretary's expert knowledge was lost, entire nodes of the organisations hierarchy were at risk of being crippled.
- With no consolidated approaches or processes for data management, handovers between knowledgeable members often went awry.

- Complex internal organisational hierarchies and sub-structures exist.
- Due to being a largely Volunteer-driven organisation, updates came from a multitude of sources. It proved to be a complex exercise to import data from such a multitude of semi-structured data sources.
- Vast geographical distribution of the localised expert knowledge was compounded by limited collaboration, leading to multiple contributors with limited visibility and collaboration.
- Lack of clear processes and guidelines.
- Lack of data validation and verification mechanisms.

The above factors lead to NGO databases that are onerous and cumbersome to maintain, largely inaccurate, and most importantly, viewed as untrustworthy by their users.

3.5 Suitability for data matching

While the data set presents numerous challenges, it can be concluded that it displays characteristics that make it ideal for analysis by a data matching system. The most important of these being that the majority of fields are discrete, meaning the fields are bounded, with a finite number of values available per field [69]. Thus the majority of DPPMS test conditions used nominal attributes and not ordinal or continuous attributes [70].

3.6 Conclusion

This chapter provided an overview of the data set, the current approaches being used for data-matching and the challenges being encountered. It was concluded that while the current update process is a source of contention for the stakeholders, the actual data set is an ideal candidate for analysis by a data matching system. The primary motivation for this conclusion is that the majority of fields within the data set are discrete with the continuous data fields largely not being required during the data matching process.

The following chapter provides an overview of the DPPMS design process.

CHAPTER 4. *DPPMS DESIGN*

4.1 Introduction

After reviewing the current approaches and problems experienced by the NGO, development of a solution termed the DPPMS was proposed. The primary objectives of the DPPMS were to accept incoming data strings, pre-process these strings, data-match them to an existing data set and then provide relevant feedback to the user.

This chapter provides a high-level overview of the design and the key components of this system. The chapter concludes with the evaluation goals of the DPPMS. A more detailed description of the implementation of the DPPMS is presented in Chapter 5.

4.2 Design overview

During the initial analysis stage, a number of key system requirements were identified. These were used to underpin all design decisions made during the design and prototyping phases. Requirements included having the ability to:

- Handle semi-structured and incomplete data
- Clean and prepare bad data
- Accurately infer missing values
- Accurately detect and resolve character-level issues in fields
- Accurately match input data to existing data at the field and record level
- Handle complex organisational hierarchies and sub-structures
- Provide relevant feedback to the end-user

Based on the above requirements, the following guiding principles were used in design:

- Interaction from expert users is preferred.
- The DPPMS should be semi-automated.
- The DPPMS should be risk averse. If there is an unsatisfactory level of confidence with a match between two fields, the match should fail.
- To ensure adequate match confidence, all partial matches, inferences and corrections to fields should be corroborated by at least one other field.
- The DPPMS should not make any unnecessary changes to data.

4.3 Process overview

The overall process followed by the DPPMS can be summarised into the following steps:

1. A string is accepted, since a single string is the one format that any of the varied sources can easily be converted into, and tokenized.
2. The canonical form is obtained for each token. This includes standardization of abbreviations and formats as well as the removal of invalid and special characters.
3. Each token is then compared to every available field, in order, in an attempt to identify a relevant field type for the token. This continues until all matches have been found or all possibilities are exhausted. If a token is matched to a field, the token is assigned a score of 2; if no match is found, the token is assigned a score of 0. Data sources used during this comparison step include:
 - a. Domain-specific keyword lists (e.g., job titles, roles and abbreviations).
 - b. Domain-specific lists of common misspellings.
 - c. The current data set.
4. The matching process above is then repeated, this time using the Damerau–Levenshtein distance metric to assist with identifying any matches obscured by misspellings or erroneous characters. The same three data sources from above are used. If a token is matched to a field, the token is assigned a score of 1, not 2.
5. If an identifier (name or role) has been confirmed as identified, this is used to infer values from the database. If inference is able to positively identify a token:
 - a. If that token has a current score of 0, it will be increased to 1.
 - b. If that token has a current score of 1, it will be increased to 2.
6. The resulting tokens are then evaluated to verify if the input string as a whole is usable. If not, it will be discarded and the relevant reason provided, viz:
 - a. Key fields are missing, for example a name or unique identifier has either not been found or has a score of 1 or less.
 - b. No tokens, other than name, have a score of 2.
 - c. Rule failure occurs. The following chapter contains further details on:
 - i. A business domain rule failure, e.g., a Volunteer attempting to fulfil a position reserved for permanent Employees.

- ii. A structural rule failure, e.g., an entity having an incorrect parent, such as a supplied suburb not being within the supplied province.
7. If the input is valid, the relevant one of 3 potential update scenarios is identified:
 - a. A positive record match is made but the data provided matches existing data. The string will be flagged as not requiring a data set update.
 - b. A positive record match is made and there is a difference between the input and the data in the current data set. The string will be flagged as requiring a data set update (amended or additional data being provided).
 - c. A positive record match is made but the hierarchical context of the input string is different to that of the record in the current data set. If all node and leaf data provided in the input string is valid, the DPPMS will assume that it is the user's intention to relocate the record within the hierarchy. The string will be flagged as requiring user confirmation.
 8. The user is informed accordingly and the corresponding action taken (viz. update occurs, or no update occurs, or update occurs after user confirmation).

These steps are shown in Figure 3 below:

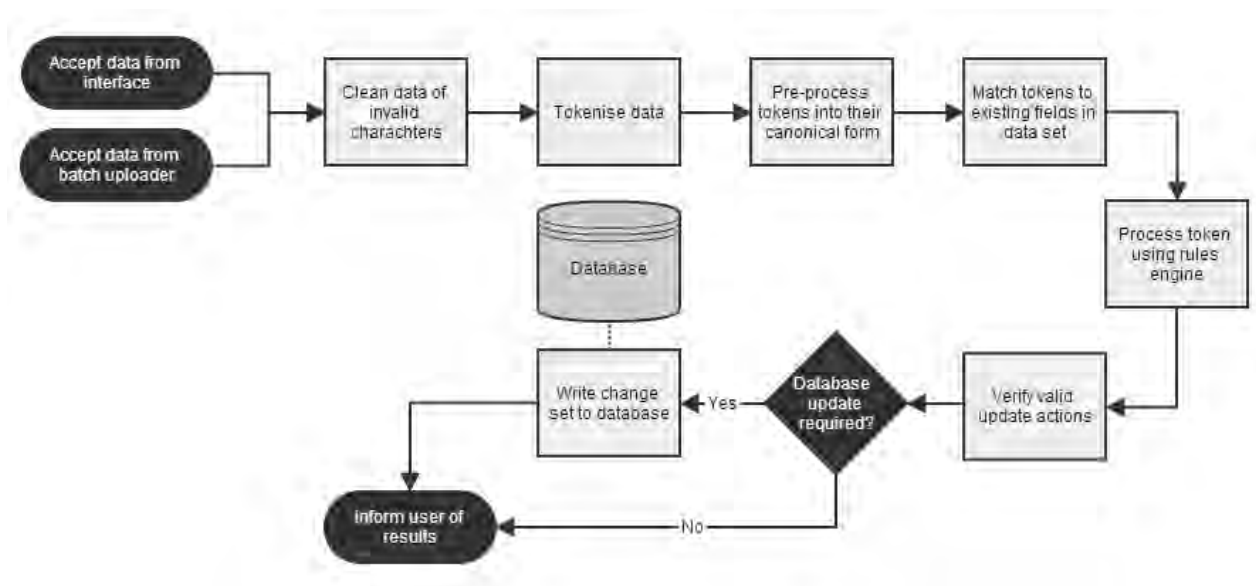


Figure 3. High level process overview.

4.4 Component overview

The DPPMS was separated into six key components:

- **Database:** the DPPMS uses a MySQL database to store existing data, current data, temporary input data, stored procedures and result data.
- **Front end interface:** This web-based interface enables the users to upload either CSV files or individual input strings, review the status and result of the import process, as well as view the data in the existing data set.
- **Data Cleaning Engine:** Each input string is tokenised and converted to its canonical form.
- **Data Matching Engine:** The data matching engine houses all field-level matching and scoring algorithms.
- **Rules Engine:** The rules engine takes the cleaned and matched input strings and applies the domain specific and hierarchal rule sets.
- **Execution and Reporting layer:** the DPPMS has a dedicated reporting and execution layer that commits required updates to the database and then presents users with the results of each input string processed.

4.4.1 Interface

From the outset, an online interface was envisioned that would enable users to interact with the DPPMS to upload their data as well review existing data. A total of five screens were implemented as part of the DPPMS, these included a home page, single string upload, bulk CSV upload, data review screen and a support screen.

4.4.2 Database

The DPPMS utilises a relational database to store existing data, stored procedures, temporary cached import data, imported data, finalized data and audit data. While not ideal for dealing with complex hierarchical data, a relational database was selected as the underlying database due to its ease of implementation, scalability, ease of integration with Java and GlassFish as well as for its flexibility with handling stored procedures.

4.4.3 Data Cleaning

All input data is tokenised, cleaned then converted to its canonical form by a dedicated cleaning engine. The primary objective of the data cleaning component was to maximise the data match potential and data match confidence while minimising the number of changes made to the input string. Data cleaning was largely a static process completed using predefined lists of acceptable and illegal characters.

4.4.4 Data matching

The data matching engine uses field-level and record-level matching logic to match:

- All tokens within the input string to field types from the existing data set.
- The input string, as a whole, to a record within the existing data set.

4.4.4.1 Field level matching

A two-phase approach was taken to data-matching [71]. Initially, an iterative process was used to identify all possible matches. Once all potential matches were identified, a validation process was used to verify and further infer other potential matches.

A loop is used to compare each input token to every field in the existing data set. The initial loop attempts to locate only positive 1-on-1 matches. The same loop is then run again, using a Damerau–Levenshtein distance metric. Care had to be taken during the design phase as the Damerau–Levenshtein distance metric is susceptible to false matches when comparing short strings. For example, a single character change to a three character string is highly likely to generate a false match while a single character change to a ten character string is less likely to. Mitigation of this phenomenon is explored further in Chapter 5. The use of a distance metric such as Damerau–Levenshtein was selected over lexicographical alternatives such as the internet based WordNet service as very few of the fields are English dictionary based nouns. The majority are proper nouns.

All tokens were then compared to two pre-defined word lists. The domain keyword list was compiled by expert users and contained all terminology and definitions that could be expected in valid input. The common misspellings list was compiled based on observation during the data input process.

The final inference step assists with providing confidence and validity for any partially matched fields, such as those matched using the Damerau–Levenshtein distance metric.

4.4.4.1.1 Scoring

A scoring system (0 for unmatched, 1 for potential match using the Damerau–Levenshtein metric, 2 for direct match to data set) indicated initial match confidence. The score for each token could be adjusted as further potential matches were discovered. During inference, an inferred token’s score was increased by 1. At the end of the data-matching phase, only tokens with a score of 2 or more were presumed to have been accurately matched and were then assigned their relevant field types.

4.4.4.2 Record level matching

Record level matching involved the matching of a person in an input string to an individual in the existing data set. This was done using identifiers such as name. By default, the DPPMS did not add any new records nor change unique identifiers of any existing records in the database. If the data matching component could not establish a positive record-level match, the record was not actioned and the user was informed. It was considered safer for users to add new records or change identifiers explicitly, since populating the database with duplicate or erroneous individuals would be a significant problem. The data matching process is summarised in Figure 4.

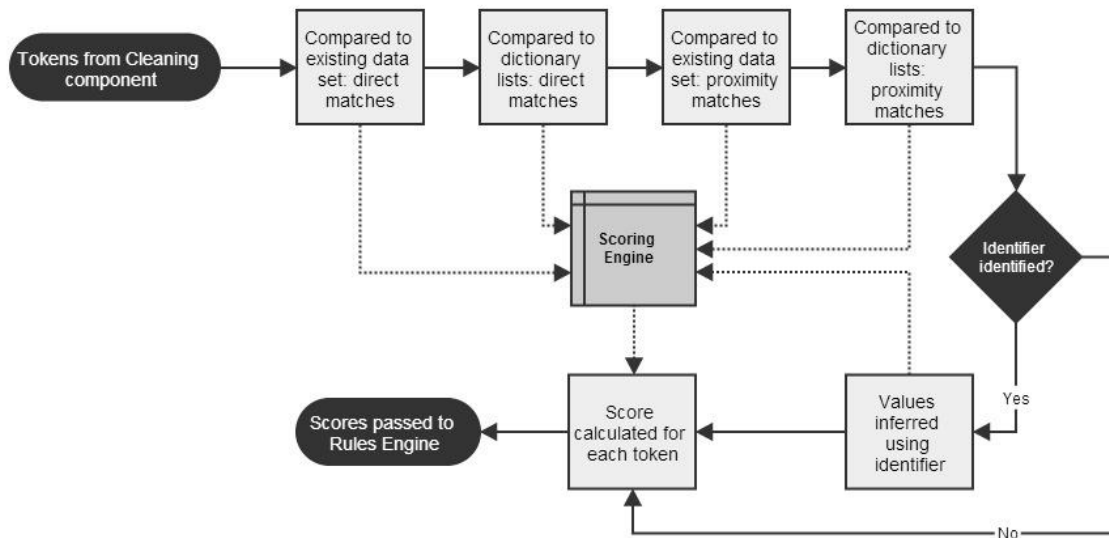


Figure 4. Data matching steps.

4.4.5 Rules

The rules were formulated based on feedback from expert users and the NGOs published internal standards and policies. A president and secretary acted as the experts during the development and evaluation of the DPPMS. The rules can be divided into four groups:

- Strict hierarchy management
- Graph relationship management
- Sub-hierarchy management
- Field value management

In working with experts, the following was noted in managing these relationships:

- The development and evaluation phases should be scheduled in advance, to avoid any significant events that will affect the experts or data. Work on this dissertation was affected by two major data improvement drives by the NGO.
- Processes followed by the DPPMS should mimic those of the stakeholders as closely as possible.
- To ensure stakeholders remain supportive, it is imperative they understand that the rule formulation and validation stages can be onerous, especially initially.

4.4.5.1 Hierarchical relationships

These are rules that relate to one-to-many relationships and represent typical hierarchies:

- A Church can have any number of Volunteers, but each Volunteer must be associated with exactly one parent Church. We denote this 1:Many relationship by \rightarrow (Volunteer \rightarrow Church).
- All Districts must have exactly one parent National Council.
- All National Councils must have exactly one global council as a parent.
- All Circuits must have exactly one parent District.
- All Churches must have exactly one parent Circuit.
- All Activities must have exactly one parent Church.

Hence, the full hierarchy can be represented as:

- Volunteer \rightarrow Church \rightarrow Circuit \rightarrow District \rightarrow National Council \rightarrow Global Council
- Activity \rightarrow Church \rightarrow Circuit \rightarrow District \rightarrow National Council \rightarrow Global Council

4.4.5.2 Graph relationships

In certain relationships an entity has a single parent which can be any one of several alternate types. We denote this by \rightarrow^1 {possible parent types}. In the NGO, the majority of these graph relationships are underpinned by the business rule that all individuals within the hierarchy have to be either paid **Employees** or unpaid **Volunteers**.

The following NGO rules involve such relationships:

- Committees can have an Event, Activity, Church, District, Circuit or National Council as their parent.
- Events can have an Activity, Committee, Church, District, Circuit or National Council as their parent.
- A Volunteer can only be associated directly to an Event, Committee, Activity or Church.
- A Volunteer cannot be associated directly to a National Council, Circuit or District.

- Each Employee must be associated to at least one parent entity; National Council, Circuit, District or Church.
- An Employee can further be associated with an Event, but not as their primary parent entity.

These relationships can be represented as follows:

- Committee \rightarrow { Event, Activity, Church, District, Circuit, National Council }
- Event \rightarrow { Activity, Committee, Church, District, Circuit, National Council }
- Employee \rightarrow { Church, District, Circuit, National Council }

There are also many-to-many relationships, which we denote using \leftrightarrow

- Activities can involve multiple Volunteers, and a Volunteer can be associated with any number of Activities.
- Committees can involve multiple Volunteers.
- Events can have multiple Volunteers and Employees associated with them.

These relationships can be represented as follows:

- Activity \leftrightarrow Volunteer
- Committee \leftrightarrow Volunteer
- Event \leftrightarrow Volunteer
- Event \leftrightarrow Employee

Certain associations are not permitted, which we denote using: \nrightarrow . For the NGO these are:

- An Employee cannot be associated with an Activity or Committee.

These relationships can be represented as follows:

- Employee \nrightarrow Activity
- Employee \nrightarrow Committee

4.4.5.3 Sub-hierarchy constraints

The existence of both hierarchical and graph relationships requires that sub-hierarchies mustn't conflict and must remain in congruence; this is enforced by rules that ensure consistency within the hierarchy. If an entity is associated with another entity, then they

must either have the exact same ancestry (belong to the same hierarchy) or else all the ancestors of one of these entities must also be ancestors of the other entity (belong to the same sub-hierarchy). An example of the former case is where a Volunteer is associated with a Committee, and both the Volunteer and Committee are part of the same National Council, District, Circuit and Church. An example of the latter is where a Volunteer is associated with a Circuit Event. This Event will only have Circuit, District and National Council ancestors - but these must be the same as the Circuit, District and National Council ancestors of the Volunteer.

4.4.5.4 Field value constraints

Certain scenarios impose uniqueness and other constraints on individual field values; these are enforced by rules dedicated to enforcing attribute value correctness:

- All records must contain a name or other unique identifying attribute.
- An Employee cannot fulfil a role designated as “Volunteers only”.
- A Volunteer cannot fulfil a role designated for “Employees only”.
- Roles for participants in Events and Committees, such as presidents, secretaries and treasurers, are not mandatory; however, where this is absent, “general participant” is injected to assist with maintaining database consistency.
- A Volunteer cannot fulfil more than one role within the same parent, i.e., Church, Activity, Committee or Event.
- An Employee cannot fulfil more than one role within the same parent, i.e., National Council, District, Circuit, Church or Event.

Further details on how these rules were implemented and enforced can be found in the following chapter.

4.4.6 Reporting

The DPPMS has a dedicated reporting and presentation layer that presents users with the results of each row or line processed; this can be reviewed immediately by the user or later via the View data screen within the interface. For advanced analysis via custom querying, more detailed results are written to a dedicated auditing table.

4.5 Evaluation goals

The primary evaluation goal of this research was: To what extent did the pre-processing of the semi-structured data assist with the data matching process without leading to the loss or change of useful information?

This was complemented by the analysis of:

- True Positives
- False Positives
- True Negatives
- False Negatives

The above values were further evaluated in terms of:

- An analysis of Precision, Recall and F-Measure for the metrics above.
- If there were any gains in accuracy and efficiency due to the expanded data pre-processing.
- Were there any risks introduced by the expanded data pre-processing steps?

This research used a multi-part approach to testing and evaluation, including rule validation, sub-sampling and test case execution.

4.5.1 Rule validation

When validating rules, the key quality indicators include consistency, completeness and correctness [31]. All rules were validated for correctness and appropriateness by relevant expert users.

4.5.2 Record sub-sampling

During development of the DPPMS, sub-sampling was used as part of a white box approach to system testing and rule validation. This was not pursued beyond the DPPMS

development phase and did not form part of the evaluation or results analysis phase of this research.

4.5.3 System level testing

Although field and record level testing was conducted, evaluation was primarily based on system level testing. During this testing, an oracle data set was compared to a data set that had been processed using the DPPMS and a data set that had been matched using the NGOs existing process. The results of this comparison were then analysed.

4.6 Conclusion

This chapter introduced the primary objectives of the DPPMS; these being to accept the incoming data strings, pre-process these strings, data-match them to an existing data set and then provide relevant feedback to the user. An overview of the overall system processes, the components deployed as well as the evaluation goals were then introduced. It was concluded that the primary evaluation goal would be to assess to what extent did the pre-processing of the semi-structured data assist with the data matching process without leading to the loss or change of useful information.

The following chapter expands on the implementation of the DPPMS.

CHAPTER 5. *DPPMS* IMPLEMENTATION

5.1 Introduction

This chapter provides an overview of the implementation of the DPPMS.

5.2 Software applications

The DPPMS was implemented using the following software components:

- **MySQL database:** Community edition version 5.5.29-0
- **Oracle GlassFish application server:** Open Source Edition 3.1.2.2 (build 5)
- **Oracle Java:** Java SE 7
- **PrimeFaces JSF implementation:** Community edition version 3.5
- **Ubuntu Server edition:** 12.04.3 LTS

While a relational database is by no means best for dealing with complex hierarchical data, this drawback was mitigated by the benefits gained such as ease of implementation and ease of integration with Java and GlassFish.

Being a Web-based application; Oracle GlassFish Server was selected as the DPPMS's Web application server. This enabled rapid and stable Web development using the PrimeFaces framework, which is based on the JavaServer Faces (JSF) specification.

The core decision and rules logic was implemented using Java. For resource intensive steps, a hybrid approach consisting of Java and stored procedures was selected to keep the business logic with the data wherever possible, and to shift the processing burden away from the Java application.

5.2.1 MySQL

5.2.1.1 Tables

The database contains 14 relations as shown in the Appendix:

- A relation for each key entity type, viz. **National Councils, Districts, Circuits, Churches, Activities, Events, Committees, Volunteers and Employees.**
- **Batch:** An operational table used when importing large data sets.
- **Audit:** An operational table used to record and monitor the outcome of all SQL queries.

- Three tables used for mapping between Roles and Activities.

5.2.1.2 SQL queries

A limited number of SQL queries were run directly from the Java code. This is not an ideal approach so was only implemented for non-resource intensive queries that are run infrequently e.g., queries used initially to construct initial data dictionaries from the existing database.

5.2.1.3 Stored procedures

The 27 stored procedures of the DPPMS were divided into two core groupings: *resolvers* used as part of the inference process and *updaters* used to update the database and interface. Due to the differing nature of Volunteers and Employees, these were subdivided to form four groupings: Employee-resolvers, Employee-updaters, Volunteer-resolvers and Volunteer-updaters. With the resolvers, the most commonly encountered issue was error and exception handling. Due to the high volume of iterations and the number of queries generated, a large number of queries returned null values. Exception handling was implemented directly in Java.

The primary function of resolvers was to infer any missing values and then validate each proposed update. That of updaters was to write each result to the database and assist with populating the PrimeFaces DataTable components within the interface. The updaters largely contained straightforward SQL statements, while also triggering audit table updates (used during DPPMS evaluation and analysis).

5.2.2 Java Code

5.2.2.1 Java classes

The Java functionality resides in six main classes while a further 24 classes supported the core functionality in the form of backing beans, objects and controllers. The batch and single entry functionality shared the same code base; the only difference was that the batch components were encapsulated in a loop for processing multiple records.

A summary of the key classes is included below:

- The following classes were backing beans and objects for the PrimeFaces DataTable components: *Activitycls.java*, *Churchcls.java*, *Circuitcls.java*, *Committeecls.java*, *Districtcls.java*, *Eventcls.java*, *Membercls.java*, *Nccls.java*, *Rolecls.java*, *Staffcls.java*, *Batchcls.java* and *TableBatch.java*.
- The following classes were used by the single string processing functionality: *DataBean.java*, *ProcessBean.java* and *ProcessempData.java*
- The following classes were used by the batch upload processing functionality: *FileUploadController.java*, *ProcessBeanBatch.java* and *ProcessempDataBatch.java*.
- *Connect.java* was a shared class used to assist with managing the Java Database Connectivity (JDBC) connections and to ensure the efficient use of the connection pool
- The following classes were used as backing beans for the report download functionality on the *View Data* page: *DownloaderBean1.java* and *Imgmgr.java*.

5.2.2.2 Java code summary

The three primary classes are *DataBean.java*, *ProcessBean.java* and *ProcessempData.java*. The *DataBean* class is responsible for accepting the string from the Web interface and then completing all pre-cleaning and tokenization tasks. This class works in isolation and does not query the database.

Once the *DataBean* class has completed and returned a result that was viable for further processing, the processed string was handed over to the *ProcessBean* class for further processing.

The *ProcessBean* class takes an iterative approach to each token; firstly, it attempts to match each token to an existing field within the database. It then attempts to infer any missing values and to create a proposed update. Finally, the proposed update is checked and the relevant results returned. The *ProcessempData* class was used by the *ProcessBean* class to assist with the complex task of modelling the potential Employee update scenarios. A dedicated Employee handling class was used due to the complexity of the Employee hierarchies and the potential for Employees to have a parent at any level

within the hierarchy. Both the *ProcessempData* and *ProcessBean* classes made extensive use of stored procedures to query the database.

The batch processing functionality had a similar code flow to the single line processing classes except that data was injected via the *FileUploadController* class and not via the Web interface. The end result of the batch process was also not outputted to the Web interface line by line but rather a summarised set of results was displayed with a detailed result set being stored in the database.

5.2.2.3 Data storage within Java and GlassFish

To assist with flexibility and scalability, the database was used to store the majority of properties and variables. The only exceptions to this were word lists provided as CSV text files and automatically uploaded into ArrayLists when the application initialised, and a temporary location */var/temp/*, on the host file system, to temporarily store batch CSV input files uploaded by users. All other data was stored in the MySQL database.

5.3 Interface

The interface consisted of eight screens; five were fully implemented while three were partially implemented.

5.3.1.1 Securing the interface

A small number of users raised trustworthiness of the DPPMS as a concern. Their primary concern was around accessibility of the interface via the internet. They were mainly Volunteers not overly proficient with IT. These were valid concerns that would need to be addressed if such a system was implemented in a production environment.

Due to the DPPMS being Web-based, steps were taken to secure the interface as well as the code and data base. The two primary mechanisms used were a Secure Sockets Layer (SSL) certificate and application level authentication. An SSL certificate was procured from a third party and imported into the GlassFish keystore.

The server was then configured to direct all HTTP and HTTPS TCP/IP traffic onto the HTTPS port (8181). The use of the SSL protocol was a best effort attempt to ensure that

data in transit was encrypted. Application-level authentication was implemented using GlassFish's Security File realm authentication functionality. This enables developers to create secure realms within their applications by specifying authentication constraints at the folder and file level. The DPPMS was implemented with one open area and one secure area.

5.3.1.2 Screen overview

The five fully implemented and three partially implemented screens are outlined below but being beyond the scope of this work, are left for future work.

5.3.1.2.1 Welcome screen

The home or *Welcome* screen is a simple landing page for the Web application.

5.3.1.2.2 Automatic - Single Row Entry screen

The *Automatic - Single Row Entry* screen enables users to upload a single string or piece of data into the text box provided, as shown in Figure 8.

The user then clicks the **Submit** button and the DPPMS returns the tokenised string. After the **Process** button is selected, the DPPMS processes the data and returns the relevant response to the user, as shown in Figure 9.

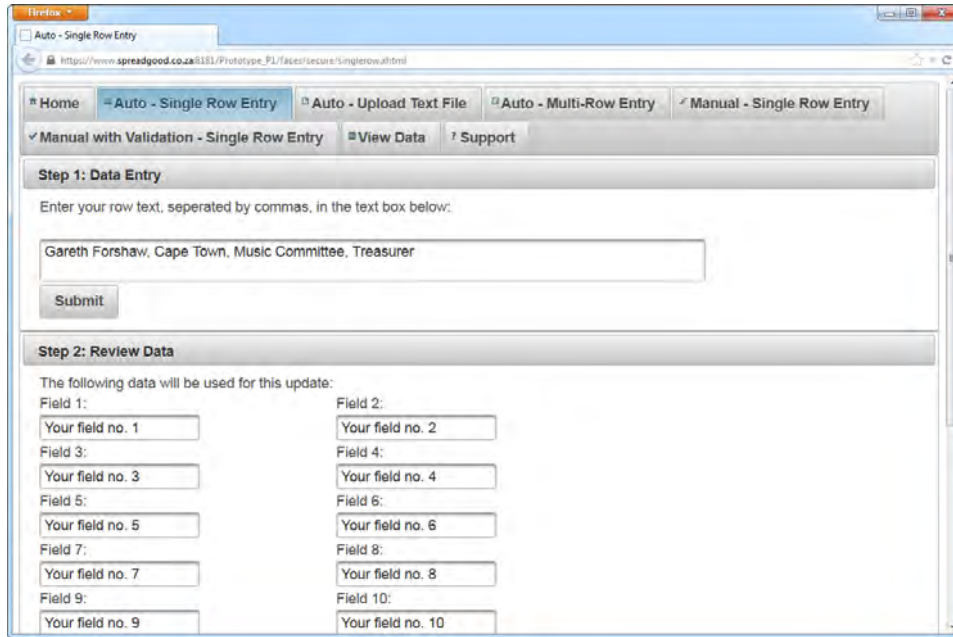


Figure 5. The DPPMS *Automatic - Single Row Entry* screen – initial user input.

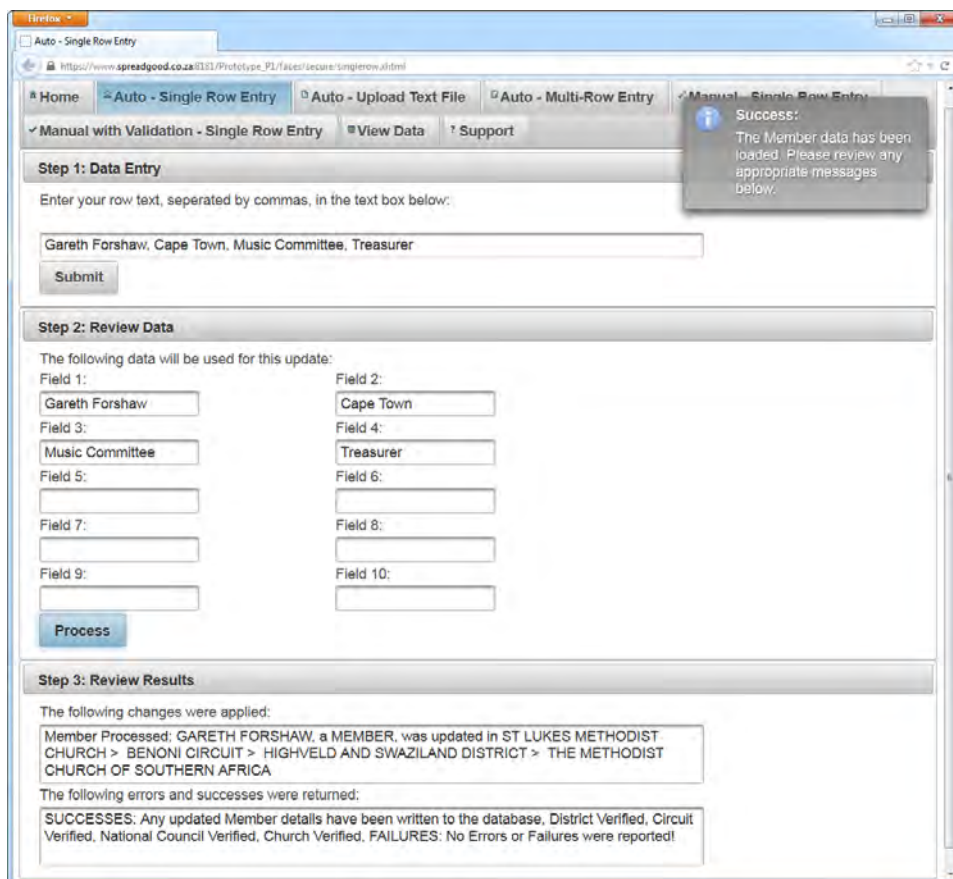


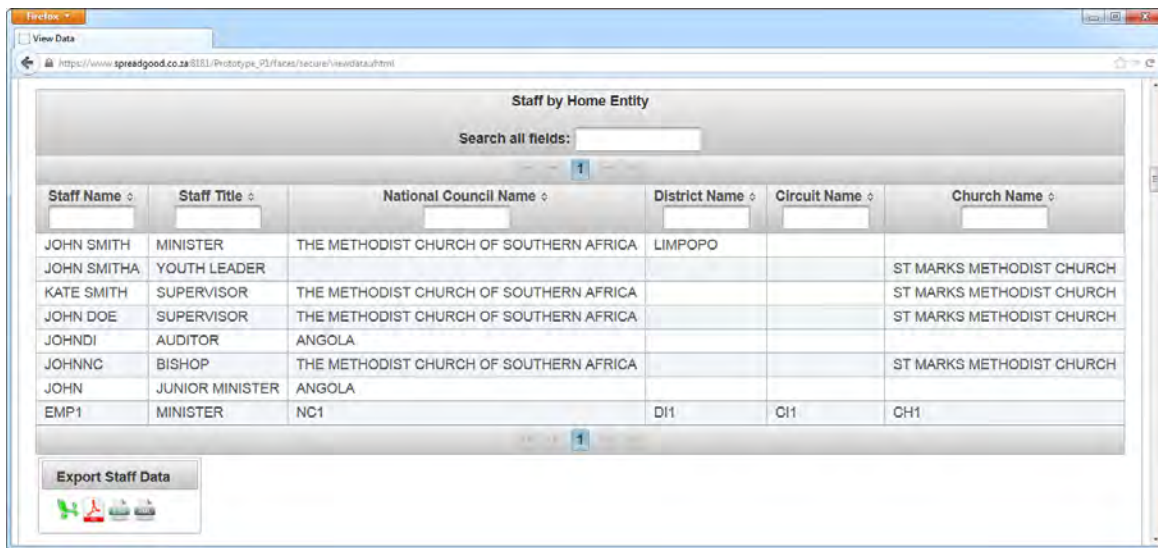
Figure 6. The DPPMS *Automatic - Single Row Entry* screen – final response.

5.3.1.2.3 Automatic - Upload Text File screen

The *Automatic – Upload Text File* screen enables users to browse their machine for the required CSV file. After the required file had been uploaded, the user selects the **Process** button to start the batch processing.

5.3.1.2.4 View Data screen

The View Data screen contains eleven PrimeFaces DataTables that enable users to review all member, Employee, Church, District, Circuit, National Council, Event, Activity, Committee, role as well as batch upload result data. These tables all include search and filter functionality, and the option to download a report in MS Excel, PDF, CSV or XML.



The screenshot shows a web browser window titled "View Data" with the URL "https://www.spreadgood.co.za:8181/Prototype_2/primefaces/secure/viewdata.html". The main content area is titled "Staff by Home Entity" and features a search bar labeled "Search all fields:". Below the search bar is a DataTable with the following columns: Staff Name, Staff Title, National Council Name, District Name, Circuit Name, and Church Name. The table contains several rows of data, including staff members like JOHN SMITH, KATE SMITH, JOHN DOE, JOHNDI, JOHNNC, and JOHN, as well as an employee EMP1. At the bottom left of the table, there is an "Export Staff Data" button with icons for Excel, PDF, CSV, and XML.

Staff Name	Staff Title	National Council Name	District Name	Circuit Name	Church Name
JOHN SMITH	MINISTER	THE METHODIST CHURCH OF SOUTHERN AFRICA	LIMPOPO		
JOHN SMITHA	YOUTH LEADER				ST MARKS METHODIST CHURCH
KATE SMITH	SUPERVISOR	THE METHODIST CHURCH OF SOUTHERN AFRICA			ST MARKS METHODIST CHURCH
JOHN DOE	SUPERVISOR	THE METHODIST CHURCH OF SOUTHERN AFRICA			ST MARKS METHODIST CHURCH
JOHNDI	AUDITOR	ANGOLA			
JOHNNC	BISHOP	THE METHODIST CHURCH OF SOUTHERN AFRICA			ST MARKS METHODIST CHURCH
JOHN	JUNIOR MINISTER	ANGOLA			
EMP1	MINISTER	NC1	DI1	CI1	CH1

Figure 7. The *View Data* screen – showing the Staff by Home Entity DataTable panel.

5.3.1.2.5 Other screens

A *Support* screen was included to assist the author and experts during development and testing. Screens for *Automatic multi-row entry*, *Manual single row entry* and *Manual single row entry with validation* were partially implemented, since interface design is outside the scope of this dissertation.

These screens were respectively for: bulk upload by pasting text into an Input text box rather than using CSV files; uploading structured rather than semi-structured data (not completed as it bore little value) and providing a real-time and interactive verification

experience to the user. All interface errors were trapped and handled by GlassFish's default error handling framework. While system-based errors were rare; user-based errors were more frequent and included supplying incorrect user details and typographical errors when entering the URL.

5.4 Reporting and Auditing

PrimeFaces provides a number of powerful message handler components. The two selected for the interface were onscreen Growl messages and onscreen Output text boxes. The *View Data* screen contains eleven PrimeFaces DataTables that enable users to review member, Employee, Church, District, Circuit, National Council, Event, Activity, Committee and role data, as well as batch upload result data. These all include search and filter functionality. The screen was also equipped with report download functionality that enables users to download a report in MS Excel, PDF, CSV or XML format.

The above methods proved sufficient for both expert usage and DPPMS system testing. MySQL Workbench 6.0 CE was the primary tool used to interrogate the database tables. Reporting is supported by the *ProcessBean* Java class as well as by the PrimeFaces DataTables, Growl and Output text components. The DataTables used to display the data were supported by the relevant backing beans and stored procedures.

5.5 Detailed system implementation

The following section outlines the implementation of each sub-process.

5.5.1 Tokenization

Implemented via PrimeFaces, the FileUploader component is used to handle the batch upload process while an Input text box is used to capture the single line input strings. Fields are tokenized out from the initial input string based on a pre-defined list of acceptable delimiting characters. This is supported by the *DataBean* Java class.

5.5.2 Data cleaning

Each field is cleaned and converted to its canonical form using a pre-defined list of common issues and anomalies, including:

- Standardization of abbreviations such as “crnr”, “St”, “Chrch” and “Dstrct”.
- Removal of invalid and illegal characters.
- Standardization of specific field formats, such as dates (MM/DD/YYYY) and phone numbers (+27 <area-code> <number>).

This is supported by the *DataBean*, *ProcessBean* and *ProcessempData* Java classes as well as via stored procedures.

5.5.3 Field-level matching

After data cleaning, each token is handed over for field matching. During field matching, each token is also scored (0 for unmatched, 1 for requiring further validation, 2 or more if identified) within the *ProcessBean* Java class.

As part of the field matching process, each token is compared against three data sources:

- **Keyword list:** this is based on the organization’s current hierarchy and contains terms like approved role names and role statuses.
- **Common misspelling list:** a dictionary of commonly misspelled words, for example, “employe”. This was formed with the aid of expert users, by general observation of the input data and by reviewing false negatives during development, where a post editing approach was taken and lists were updated based on prevailing error patterns and individual results [72]. To ensure consistency, lists were not modified after the final round of evaluation started.
- **Existing data source:** All tokens are compared to the existing data set twice, firstly using a 1-on-1 algorithm to identify identical matches and then again using the Damerau–Levenshtein distance metric. Due to the maximum distance threshold value being skewed by short input strings, a secondary percentage method was implemented. For each potential match that was located, the percentage of change (PC) required was calculated based on the input token

character length and the actual number of character changes required:

$$\text{Percentage Change (PC)} = \frac{\text{Required character changes (DI)}}{\text{Total token length (TL)}} * 100$$

Due to the differing nature of the fields, different percentage thresholds were selected for different entities. The default threshold used was 20%, hence, only tokens requiring character changes of 20% or less were considered as potential matches. The Church, District and Circuit entities had a number of similar names that resulted in false positives. To overcome the high similarity of these naming conventions, a threshold of 10% was used for these entities. These percentages were selected by analysing the acceptable risk of false positives versus false negatives. The addition of the percentage method effectively mitigated the number of false matches caused by tokens that were under five characters in length. Potential matches in this step were scored lower compared to the matches in the other steps as these were not deemed to be positive matches unless another field was able to validate the match.

After field matching, a secondary field identification loop is triggered that attempts to use any identifier (name or role) to mine the database for inferred values. If a name field has been identified, this is injected into a stored procedure to try and ascertain that individual's current type and position within the hierarchy. If a role field has been identified, this is injected into a stored procedure to ascertain whether the provided role is for a Volunteer or an Employee. The primary function of this step is to validate any potential matches; any inferred tokens have their score increased by 1. Thus e.g., those scored 1 (unconfirmed) would increase to 2 (acceptable).

5.5.4 Full string verification

Once all tokens have been cleaned and matched, all positively identified tokens are concatenated. This string is then validated against two possible failure conditions:

- Key fields not being present in the string, e.g., the string does not contain a name.
- A rules related failure occurs. Enforcement of rules is explored further below.

This is supported by the ProcessBean and ProcessempData classes and stored procedures.

5.5.4.1 Rules enforcement

5.5.4.1.1 Hierarchical relationships

These are enforced by ensuring that no entities are added to the data unless correctly associated to their unique parent. If that is ensured, all ancestors of the entity are also associated with the new entity. If not, an error is generated and the entire string rejected.

5.5.4.1.2 Graph relationships

These are enforced by ensuring no entities are added to the data unless associated with a unique parent and that it is of permitted type. If this is ensured, all ancestors of the entity are also associated with the new entity. If not, an error is generated and the input rejected.

5.5.4.1.3 Sub-hierarchy enforcement

The parent of each entity is stored in that entity's relation. Using this parent, the DPPMS is able to generate the path of any entity in the DPPMS. Certain paths will always follow a set pattern due to organisational rules, for example, entities such as Volunteers, Churches, Circuits, Districts, Activities and National Councils only require the use of a single stored procedure to generate their paths as they are always constrained to a path of:

- **Volunteer** - *VolunteerID, ChurchID, CircuitID, DistrictID, NCID*
- **Church** - *ChurchID, CircuitID, DistrictID, NCID*
- **Circuit** - *ChurchID, CircuitID, DistrictID, NCID*
- **District** - *DistrictID, NCID*
- **Activity** - *ActivityID, ChurchID, CircuitID, DistrictID, NCID*

Unlike the above, Employee, Committee and Event entities may have parents whose types differ. With these entities, the parent type needs to be ascertained before the relevant path can be generated. All potential parents are included below:

- **Employee** - *ChurchID, CircuitID, DistrictID, NCID*
- **Events** - *EventID, CommID, ActivityID, ChurchID, CircuitID, DistrictID, NCID*
- **Committee** - *CommID, EventID, ActivityID, ChurchID, CircuitID, DistrictID, NCID*

Figure 5 represents the logic applied when resolving the path of an Employee.

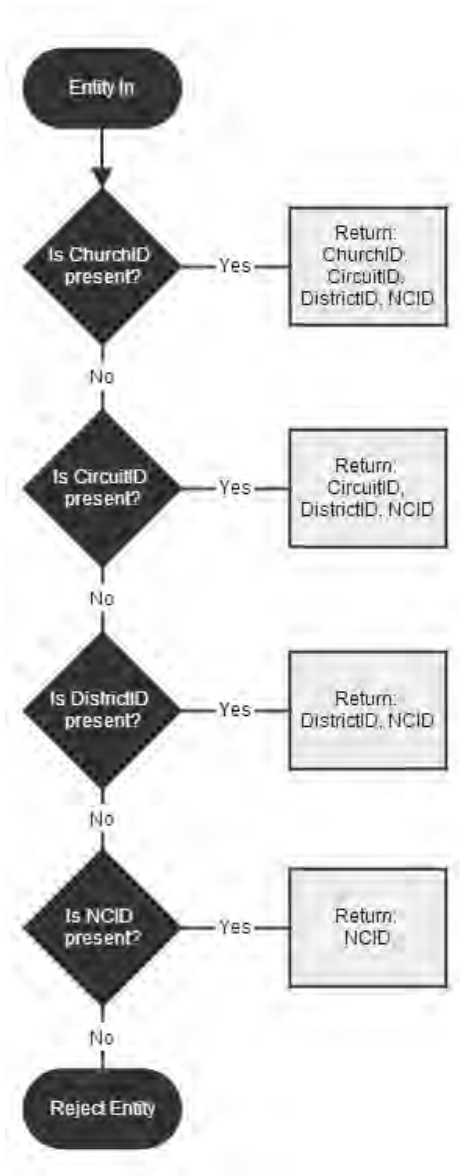


Figure 8. Path resolution for an Employee.

As each entity in the database must have a valid parent, the DPPMS will always be able to generate an accurate path; it may just require multiple iterations and database queries to accurately generate the paths.

Inference can also be used for many-to-many relationships such as:

- Activity ↔ Volunteer.
- Committee ↔ Volunteer
- Event ↔ Volunteer

- Event ↔ Employee

For example, when a Volunteer is assigned to a District level Event, the Volunteer's District must match that of the Event's parent District. The diagram below represents the process used to identify the path as well as ensure the legitimacy of the hierarchy for a Volunteer (vol) associated to an Event.

Unless one of the conditions above resolves to true, the DPPMS will reject the string based on a sub-hierarchy violation. The approach outlined above holds true for all activities, committees and events, both Volunteers and Employees. This logic is implemented via Joins in stored procedures along with Java code.

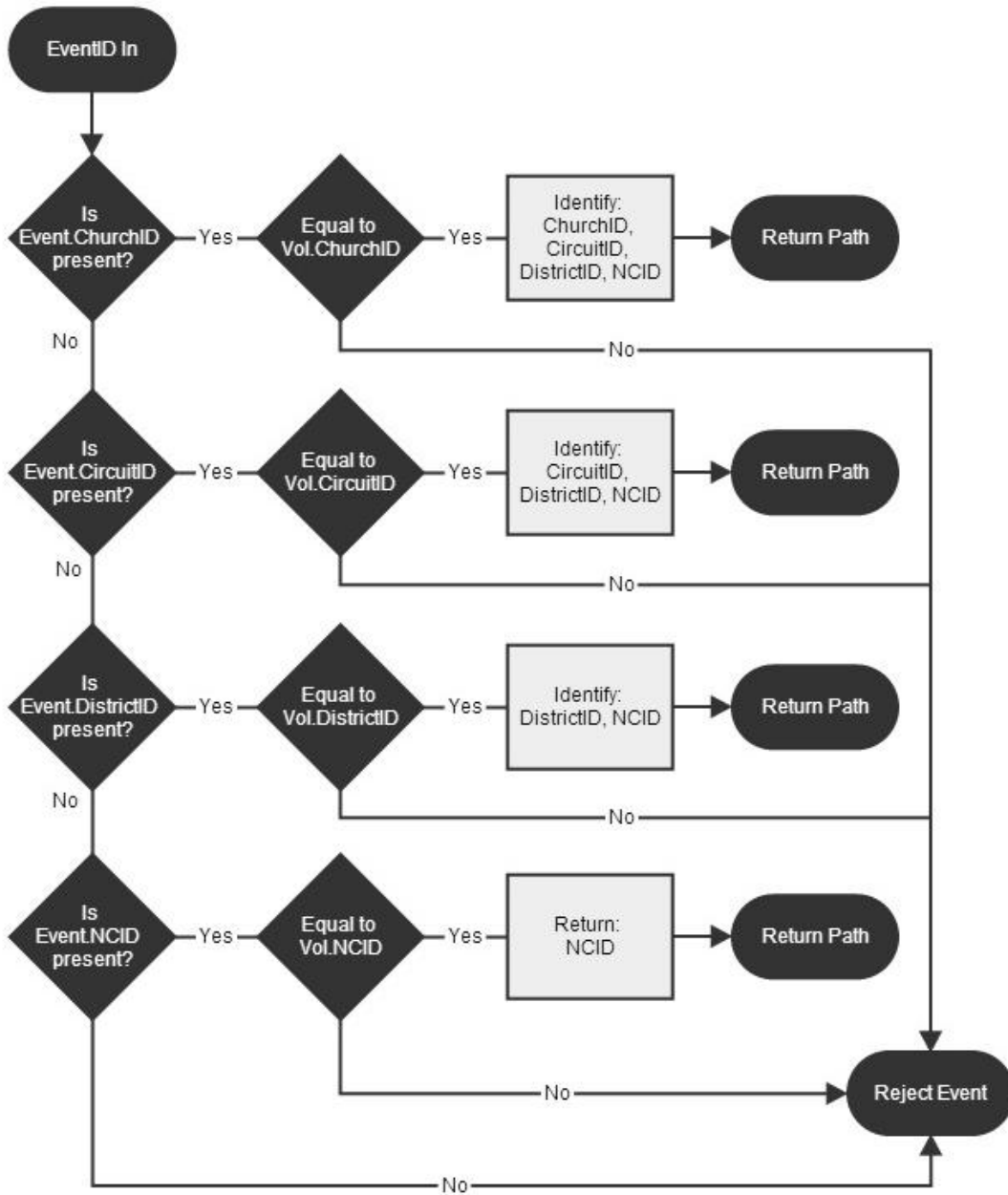


Figure 9. Event path resolution for a Volunteer (Vol).

5.5.4.1.4 Field value constraints

Uniqueness is evaluated primarily by querying the existing data set. Updates are constrained predominantly based on membership type; Volunteer or Employee.

Once membership type has been established, the DPPMS will ignore any input token that is not correct for the membership type provided. The string will not be rejected; the user will, however, be informed that a particular field was provided but not used by the DPPMS.

5.5.5 Available update actions

Once all data-cleaning, matching and rules analysis has completed, the next step is for the DPPMS to determine the required update action. There are four possible:

- *Successful – Update required:* All rules have been passed, at least two of the input tokens have been identified, and an individual referenced in the input string has been matched to an individual in the existing data. Additional or updated data has been included in the input string that can be used to update the existing data set.
- *Successful – No update required:* As above; however, no differences or additions have been identified that necessitate an update of the data set.
- *Successful - Relocation authorization required:* As above; however hierarchical path data in the input string does not match the path in the existing data set, requiring manual review prior to relocating the individual in the hierarchy.
- *Rejected:* The string, having failed one or more phases in the process, will be rejected and a relevant message returned to the user.

5.5.5.1 Updating an entities details

Once a writable update condition has been confirmed, the tokens are passed to a stored procedure and the data set is updated. Valid updates include:

- Associating a Volunteer with an Event, Committee or Activity.
- Associating an Employee with an Event, Church, Circuit, District or National Council.
- Updating a Volunteer's details.
- Updating an Employee's details.

This was supported by the *ProcessBean* Java class as well as the stored procedures. Figure 7 shows the update logic used.

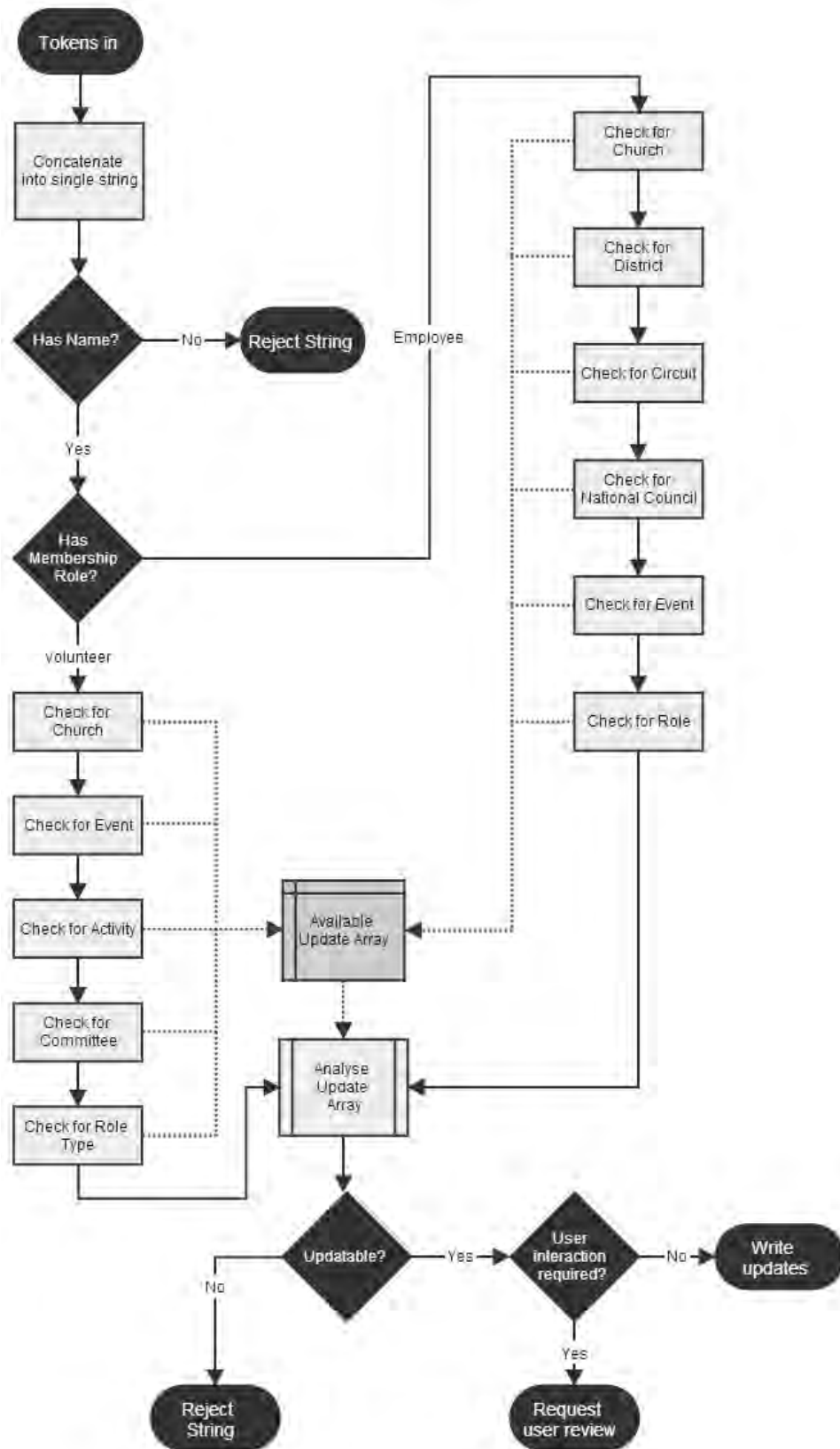


Figure 10. Final update logic.

5.6 Conclusion

This chapter took provided an in-depth look at the implementation of the DPPMS. This included a review of the Database layer, Software layer, User Interface, Reporting Functionality as well as a detailed review of the systems internal processes. The key sub-processes were identified as tokenisation, data cleaning, field-level matching and rules enforcement.

The following chapter reviews the applicability of the DPPMS to other data sets.

CHAPTER 6. APPLICABILITY TO ALTERNATE DATA SETS

After the successful development of the DPPMS for the NGO data set, an investigation was conducted into whether the DPPMS was applicable to other data sets that shared similar characteristics. The two data sets selected were patient data from the health care sector and customer data from the financial sector.

6.1 Overview of the data sets

6.1.1 Introduction to data in the health care sector

With the ever increasing government focus on service delivery and improved health care in South Africa, the accurate capturing and analysis of health care, and primarily patient, data has assumed a much greater significance recently [73]. Accurate health care data assists researchers and resource planners by allowing them to apply statistical analysis, epidemiological analysis and other disciplines to the data sets [74]. Patient data is however notoriously inaccurate [8; 27] which devalues the potential of the data sets [75]. Importing of patient data could benefit greatly from a system that could accurately and effectively pre-process and then match the data to an existing data set [76].

6.1.1.1 Data set overview

The medical data set used for this dissertation was based on 3 months of admission data for a Cape Town hospital. The data set consisted of 9000 records. This data was limited to initial diagnosis and patient data capture and did not include any follow ups or on-going treatment data. The key attributes were: Out-patient or In-patient; Referred or Walk-in; Initial diagnosis or Symptoms; Area of capture (District, Suburb, Hospital, Ward); Status (recovered/deceased); Patient identifier.

6.1.1.2 Current update process

Due to the procedures inherently in place within medical facilities, the potential for accurate data capture was the highest of the three scenarios examined in this dissertation. Unfortunately, due to a number of factors, this was not the case and the quality of data captured was poor.

All incoming patients are greeted by a nurse who captures the initial patient details, either directly into a terminal or onto a hand-written admissions card. The majority of institutions now appear to be using a direct capture approach via terminals. There are no verification or validation methods in place and data is captured “as is”.

6.1.1.3 Data storage

Data that was captured via a terminal goes through extremely limited quality control and is then imported into a Medical Information Database. Hand written data is captured by data capturers then imported into the same Medical Information Database.

6.1.1.4 Data quality concerns

A number of challenges were identified with the current patient data capture process:

- Nurses with limited exposure to information technology.
- Medical facilities that are understaffed and often overwhelmed.
- Language barriers between the patients and medical staff.
- Patients that are deliberately elusive with their details for a variety of reasons.
- Inaccurate capturing of admission forms by back office staff, often due to illegible forms or misunderstanding of the data.
- A lack of adequate back office data capture procedures and quality control processes.

Further factors contributing to the poor quality of the patient data were identified as:

- Spelling incorrect (e.g., words spelt phonetically and evidence of language barriers).
- Incorrect data placement in the form (e.g., suburbs and road names reversed).
- Use of apartment/complex names and hostel names instead of actual street addresses.
- Slang and commonly accepted, informal terms to describe an area e.g., “Waterfront” and “Barcelona”, both in the township officially known as Gugulethu.

6.1.2 Introduction to data in the Financial Services sector

Customer data of the financial services industry are considered well-structured and well governed. Government regulations such as FICA and other legislation oversee and regulate data collection and storage. While this may be true for financial data in developed countries, the realities in developing countries is often extremely different.

The growing popularity of banking innovations, such as mobile money, is creating previously unseen challenges within the banking sector [77]. The fact that the largest growth in mobile banking is occurring in emerging markets further compounds the already challenging task of accurately capturing and importing new customer data [3; 8; 78]. Furthermore, the sign-up process for mobile money account holders frequently defies the structured approach of traditional banking processes. The reason for this break in tradition is that often potential mobile money customers sign up for mobile money accounts via a field agent *or* representative of the bank or mobile network operator (MNO); they rarely sign up directly with the service provider at a traditional branch [73].

Accurately capturing the *know your customer (KYC)* data is the most critical step in the new mobile money customer on-boarding process [79]. Data quality issues are common as this data is often captured by an intermediary and then relayed to the service provider, often via hand written forms completed by the customer or the agent [80]. This data is then processed by a central processing unit, often a number of days after initial capture. This unit has no context or knowledge of the customer. More advanced systems see the agent completing these details on a handset or terminal, in near-real-time. Suffering a similar fate as the NGOs, the data capturers are often overwhelmed by inaccurate data, a multitude of data source types and a lack of knowledge experts to resolve queries [81].

Importing mobile money customer data could benefit greatly from a system that accurately and effectively pre-processes then matches the data to an existing data set.

6.1.2.1 Data set overview

For this dissertation, a financial data set was used that represented 10 000 banking clients. The data set was divided between two bank account (product) offerings with certain clients having a credit card linked to one of two credit card issuers.

The key attributes of the data were thus: Know Your Customer (KYC) data, Branch the customer registered at, Remote channel or branch walk-in, Account type, Card type (if any), Local catchment of the customer (Region, Province, Suburb, Branch), Account status (active/frozen/closed) and a unique Customer identifier.

Know Your Customer (KYC) data can be used to unambiguously identify a banking customer. KYC data capture requirements are governed by law to reduce crimes such as money laundering, tax evasion and financing illicit activities. In South Africa this is governed by the Financial Intelligence Centre Act no 38 of 2001, known as FICA [82].

6.1.2.2 Data storage

Financial data is stored in well-structured and secured data centres. Financial data is nearly always stored in a Relational Database. These enterprise grade databases are often governed by strict update procedures that ensure the integrity of all data remains intact.

6.1.2.3 Current update process

With banking products such as mobile banking, customer registrations are often carried out on a commission basis in the field by agents affiliated with a bank or mobile network operator (MNO). Data capture is done in two key ways, either in real-time by the agent entering the customer details into a handset or terminal, or more commonly, by the agent completing a hand-written form. Data captured using a hand-held device is often of a better quality as the device can perform limited online validation, such as, checking if the potential customer is already an existing customer.

Hand-written data is relayed to the relevant service provider via post, fax or directly. This data is then processed by a central processing unit, often a number of days later.

This unit has no knowledge of the customer other than what is provided on the original form. A third scenario is a hybrid approach in which agents capture customer details on

paper throughout the day and enter this via a hand-held device or terminal at the end of the day.

6.1.2.4 Data quality concerns

In the scenarios above with a two-phase data capture process, data quality issues are common. Similar to the NGO, centralised data capturers are often overwhelmed by inaccurate data, a multitude of data source types and a lack of experts to resolve queries.

Commonly encountered data issues include incomplete or missing KYC data, missing or incorrect branch details, customers attempting to re-register instead of updating their details, and incorrect mobile (cell phone) numbers being provided

Consequences of poor quality data in the financial sector include accounts being erroneously frozen, law enforcement agencies launching unnecessary investigations, missing or incorrectly routed payments and financial institutions having incorrect customer contact details.

6.1.3 Similarities to the NGO data set

While the terminology used within the three fields is dissimilar, it was evident that the data sets were structurally similar, as can be seen in Table 4.

Characteristic	NGO Data Set	Medical Data Set	Finance Data Set
Strict hierarchical relationships	National Council → District→ Circuit→ Church	Province → Magisterial District → Catchment area → Hospital	Province → Region → Suburb → Bank branch
Mandatory parent entity for individuals	Church for Volunteer or an appropriate parent for Employees.	Hospital for patients.	Bank branch for bank customers.
Graph relationships	Links to Events, Activities or Committees.	Links to Wards, clinical trials or studies.	Links to Saving schemes, investment accounts or debit/credit cards.
Different Types of Individual	Employee or Volunteer.	In-patient or Out- patient.	Private Customer or Business Customer.
Relationships not allowed	Employees cannot be linked to Activities or Committees.	Out-patients cannot be linked to Wards. Adults cannot be linked to paediatric hospitals.	Private Customers cannot be linked to a business account. Islamic customers cannot be linked to non-Sharia account types.
Status or Roles for Individuals	“Registered member”, “President” or “Treasurer”.	“Living” or “deceased”. “Admitted” or “Discharged”	“Active customer” or “Blocked customer”. “Generic banking customer” or “Islamic banking customer”.
Field value constraints	A unique personal identifier: Name.	A unique personal identifier along with critical medical data such as age and blood type.	A unique personal identifier along with mandatory KYC data as specified by government.
Mandatory time-based fields	Sign-up date.	First-treatment date.	Customer registration date.

Table 4. Similarities across NGO, Patient and Financial data.

6.2 Overview of requirements

6.2.1 Precision and Recall

As discussed, depending on the type of data being matched, either a high Recall rate or a high Precision rate may be desired.

The health and financial sectors can often face serious repercussions for erroneously identifying individuals so compromise by aiming for a higher Precision rate with a lower Recall rate. In contrast, NGOs are more likely to settle for a high Recall rate with a lower Precision rate. This was one of the primary differences between the three domains.

6.2.2 Handling of incomplete data

The approach to incomplete data differs between the three domains. Medical systems fall under the category of safety critical systems as even a seemingly minor fault in the DPPMS can have a catastrophic impact [77; 83]. Thus they are least tolerant of missing and incomplete input data. Somewhat ironically, the financial sector is the most receptive to incomplete data, which is often knowingly uploaded to the database, but placed in a suspended and non-transactional state until the missing data is provided.

6.2.3 Potential security implications

In the financial sector, concern was raised regarding the DPPMS and whether it could be “gamed” or taken advantage of by individuals with ill intentions. The sensibility and legality of automatically cleaning and matching KYC data was also raised as a potential concern. A number of industry experts were of the opinion that customer provided data should be used “as-is”, and that any form of automated correction could jeopardise the legitimacy and integrity of the data-set.

6.3 Required system updates

Based on the comparison of the data set attributes above, the DPPMS was reviewed and all required modifications noted. No major design or structural changes were identified.

The three components identified as requiring updates included the interface, the key word lists and the rules engine. These updates are explored further below.

6.3.1 Rule updates

The NGO rule set and the current rule framework were reviewed in conjunction with the rule requirements of the financial and health care data sets.

The existing rules framework was deemed to be valid as all three data sets displayed similar traits with regard to their hierarchical structures and entity constraints. The majority of rules were accommodated via updates to terminology and minor rewording and refactoring of the existing rules.

6.3.2 Dictionary lists

Common misspellings and domain specific keywords were modified for the new domains.

6.3.3 Interface updates

Updates to the application interface revolved largely around the re-alignment of terminology in on-screen text, error and success messages using PrimeFaces built-in translation and locale components.

The inclusion of the health and financial sector data sets demonstrated the flexibility of the DPPMS. The transition between these data sets was, however, a manual one with a dedicated interface being used for each separate data set. This process was not only costly but also cumbersome for end-users. A number of options for rapid reconfiguration of a generic interface were investigated, these are discussed further below.

6.3.3.1 Rapid data set selection

A prototype interface was developed with an initial landing page containing a drop-down list to select a database prior to uploading data. This ensured that the correct interface, terminology, data sets and rule sets were utilised.

6.3.3.2 Custom data set configuration

The ability to configure an entirely new data and rule set via the interface was explored. The motivation behind this idea was that NGOs are often required to configure data sets for new projects such as community initiatives. The hierarchy and associated business rules of these projects are often similar in nature to the data sets used within this dissertation.

The idea would be for the NGO to configure a new data set by completing a number of fields. The system would then generate all applicable rules, database schemas and screens. A mock-up screen was created that enabled users to specify the names of all nodes in the hierarchy, as well as mandatory and optional fields.

A number of challenges would need to be overcome to successfully enable the custom configuration of data and rule sets, these include:

- A mechanism to dynamically generate a custom database schema for each new configuration set
- Creation of a truly generic rules framework
- Development of self-learning word lists

While this would be complex to implement, the potential of such functionality is high.

6.4 Conclusion

The three data sets selected were found to share a number of common challenges. They also share a number of characteristics that make them ideal candidates for a common framework approach, the primary characteristic being that the majority of their fields are discrete. This assisted greatly during matching as the majority of test conditions took the form of nominal attributes and not ordinal or continuous attributes. The fields from all three data sets that contained continuous data were largely unused during the data matching process.

The overall effort required to accommodate the two new data sets was less than expected. The majority of the rule logic and hierarchical constraints remained intact and valid across the three data sets with the required updates largely being cosmetic. The work done in this chapter successfully transitioned the DPPMS from being a domain-specific

tool to a generic system that could be applied to a variety of data sets that featured similar hierarchies and attribute constraints.

The following chapter summarises the testing and evaluation of the DPPMS.

CHAPTER 7. TESTING AND EVALUATION

7.1 Introduction

To evaluate if pre-processing of the semi-structured data assisted with data matching without loss or change of useful information, the following were computed for a variety of test cases: Precision, Recall and F-Measure. Rule validation and sub-sampling were also used to evaluate the accuracy of the DPPMS.

7.2 Rule validation

Rule validation was completed using two approaches, an expert review and targeted white box testing. The rules outlined in Chapter 4 were presented for validation to three experts who hold senior positions within the hierarchy and are recognised as being the most knowledgeable users in the organisation. After reviewing the rules, the experts were satisfied with their design and structure.

To verify the data matching and rule checking components, an algorithm was developed to generate all possible permutations of input conditions that the DPPMS could encounter. This phase of testing exclusively targeted the rules and field matching components; data cleaning and database update did not form part of this testing phase. A VB script was used to generate all possible combinations of the ten field conditions that could be expected in an input string. The initial scenario list consisted of ten columns, the Membership Type column was populated with three potential values of <VOL>, <EMP> or <FALSE> while the remaining nine columns were populated with the placeholders <TRUE> or <FALSE>. A total of 1536 possible scenarios were identified. Using the rules established in Chapter 4 and Chapter 5, each scenario was then manually analysed and its expected overall DPPMS result recorded as either a Pass or Fail. It was expected that 1252 of the scenarios would pass while 284 would fail. Each <TRUE> placeholder was then replaced by a valid token from the target data-set while <FALSE> placeholders were populated with invalid tokens. Each scenario was then submitted to the DPPMS for processing. The actual result of each scenario was extracted from the database and compared to the predicted result.

With all scenarios returning the desired outcome, the field-matching and rules components of the DPPMS were deemed to be suitably validated.

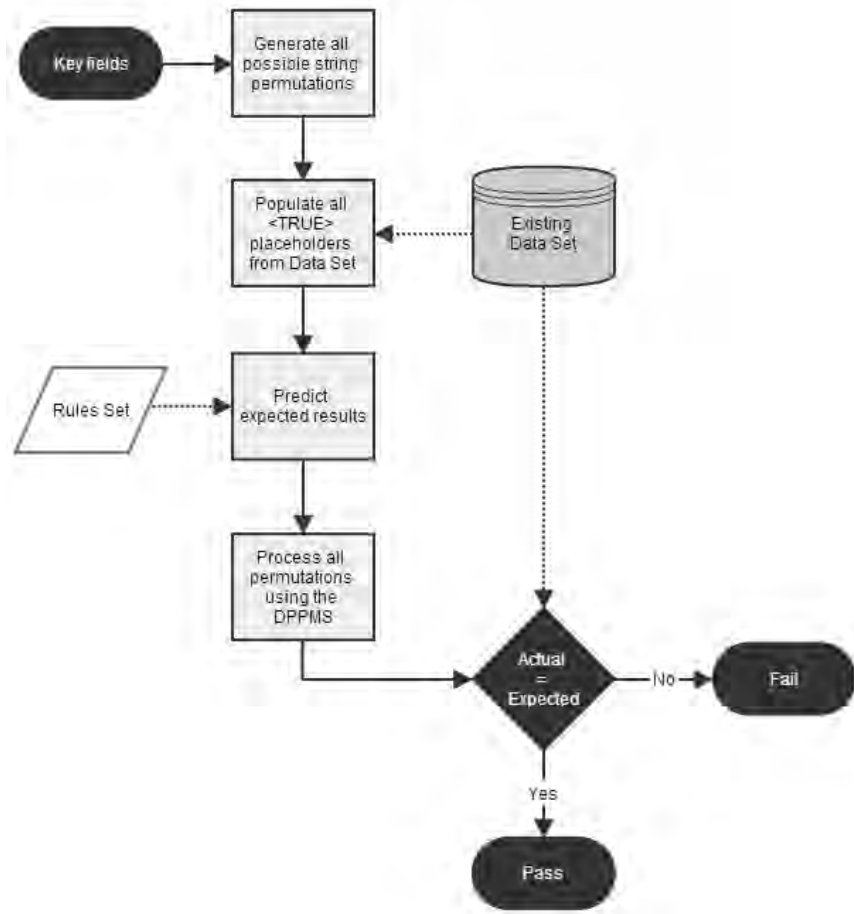


Figure 11. Overview of the rule validation process.

An extract from the initial scenario permutations table is included below, where “T” represents the <TRUE> and “F” the <FALSE> placeholders:

Name	Type	Role	Church	District	Circuit	NC	Activity	Event	Committee	Expected Result
F	F	T	T	T	F	F	T	T	T	FAIL
T	VOL	T	F	T	T	F	F	T	T	FAIL
F	VOL	F	F	T	F	T	T	F	T	FAIL
T	EMP	T	T	F	T	T	F	F	T	PASS
F	EMP	F	T	F	F	F	F	F	T	FAIL
T	F	T	F	F	T	F	T	T	F	FAIL
F	F	F	F	F	F	T	F	T	F	FAIL
T	VOL	F	T	T	F	T	T	F	F	PASS
F	VOL	T	F	T	T	T	F	F	F	FAIL
T	EMP	F	F	T	F	F	F	F	F	PASS

Table 5. Extract from the scenario permutation matrix of the NGO data set.

7.3 System-level testing

System level testing was conducted by taking a source data set and inputting it into a target data set. This process was repeated three times using the same data sets but three different input methods. The three resulting data sets were then compared. This process was repeated three times, one for each of the domains: health, finance and NGO.

The three input methods used are explained further below:

- **The Oracle:** Unprocessed input data was manually matched with assistance from the three expert users who did the rule validation. The data set was further validated using an extensive manual review. This was used as the base-line during evaluation and it was assumed that neither False Positives nor False Negatives were present in this data set.
- **The current approach:** The unprocessed input data was handled using the domain’s existing approach by an existing clerk manually matching each record. No validation or pre-processing steps were carried out; and 3 different data capture clerks were involved.
- **The DPPMS:** The input data was submitted to the DPPMS for processing.

For the financial and health care sectors, a similar approach was used to create the three data sets. Two experts assisted with creating the financial Oracle and one expert assisted with creating the health care Oracle. For the current approach, both the financial and the health care data were matched by a current data capture clerk.

For each domain, the data sets produced by the DPPMS and that produced using the current approach were compared to the Oracle data set. Metrics were then generated for each of the three domains. The results of this analysis are discussed further below.

7.3.1 Health sector

The health sector test data set consisted of a 3000 record data set with each record representing a patient visit to a hospital. This data set consisted of both in and out patients, referrals and walk-ins, unique patients as well as 270 patients that were revisits.

7.3.1.1 Health sector results

The following is a summary of the health sector test results:

	True Positives	False Positives	True Negatives	False Negatives
Oracle Date Set	2601	0	399	0
DPPMS matched	2447	2	398	153
Matched using existing approach	2369	9	252	370

Table 6. Summary of the health sector test results.

The Recall, Precision and F-Measure values were then calculated:

- **DPPMS matched:**
 - **Precision:** $\frac{2447}{2447 + 2} = 0.999183$
 - **Recall:** $\frac{2447}{2447 + 153} = 0.941154$
 - **F-Measure:** $2 * \left(\frac{0.999183 * 0.941154}{0.999183 + 0.941154} \right) = 0.969301$

- **Matched using existing approach:**
 - **Precision:** $\frac{2369}{2369 + 9} = 0.996215$
 - **Recall:** $\frac{2369}{2369 + 370} = 0.864194$
 - **F-Measure:** $2 * \left(\frac{0.996215 * 0.864914}{0.996215 + 0.864914} \right) = 0.925933$

	Recall	Precision	F-Measure
Health sector:			
DPPMS matched	0.941154	0.999183	0.969301
Matched using existing approach	0.864914	0.996215	0.925933

Table 7. Summary of the Recall, Precision and F-Measure values for the health sector.

The F-Measure, Recall and Precision rates of the DPPMS results were higher than those of the data matched using the current approach. The Precision and F-Measure values were only marginally so, unlike the Recall rate. A true positive rate increase of 2.6% was observed, the least of all three data sets.

7.3.2 Financial sector

The financial sector test data set consisted of a 3000 customer data set. This data set consisted of business and private customers, two separate card products, branch and field captured registrations, existing customers as well as 150 new customers.

7.3.2.1 Financial sector results

The following is a summary of the financial sector test results:

	True Positives	False Positives	True Negatives	False Negatives
Oracle Date Set	2810	0	190	0
<i>DPPMS</i> matched	2630	3	185	182
Matched using existing approach	2538	198	112	152

Table 8. Summary of the financial sector test results.

The Recall, Precision and F-Measure values were then calculated:

- ***DPPMS* matched:**

- **Precision:** $\frac{2630}{2630+3} = 0.998861$
- **Recall:** $\frac{2630}{2630+182} = 0.935277$
- **F-Measure:** $2 * \left(\frac{0.998861 * 0.935277}{0.998861 + 0.935277} \right) = 0.966024$

- **Matched using existing approach:**

- **Precision:** $\frac{2538}{2538+198} = 0.927632$
- **Recall:** $\frac{2538}{2538+152} = 0.943494$
- **F-Measure:** $2 * \left(\frac{0.927632 * 0.943494}{0.927632 + 0.943494} \right) = 0.935496$

	Recall	Precision	F-Measure
Finance sector:			
<i>DPPMS</i> matched	0.935277	0.998861	0.966024
Matched using existing approach	0.943494	0.927632	0.935496

Table 9. Summary of the Recall, Precision and F-Measure values for the finance sector.

As with the health sector data, the Precision and F-Measure values were more favourable for the DPPMS results. It was interesting to note, however, that the Recall rate was better for data matched using the existing approach than it was for the DPPMS matched data. This is likely to be a result of the approach of accepting incomplete data but flagging this as “suspended” in the financial sector; the DPPMS has no such mechanism for handling partial data updates. The 198 false negatives for the data matched using the existing approach is disconcerting compared to the DPPMS’s 3 false negatives. A true positive rate increase of 3% was observed.

7.3.3 NGO domain

The NGO domain test data consisted of a 3000 member data set. This consisted of Employees and members, and tested hierarchical and graph relationships, multiple roles and unique members.

7.3.3.1 NGO membership results

The following is a summary of the NGO membership test results:

	True Positives	False Positives	True Negatives	False Negatives
Oracle Date Set	2870	0	130	0
DPPMS matched	2437	1	252	288
Matched using existing approach	1794	220	90	896

Table 10. Summary of the NGO test results.

The Recall, Precision and F-Measure values were then calculated:

- **DPPMS matched:**
 - **Precision:** $\frac{2438}{2438+1} = 0.99959$
 - **Recall:** $\frac{2438}{2438+288} = 0.894351$

- **F-Measure:** $2 * \left(\frac{0.99959 * 0.894351}{0.99959 + 0.894351} \right) = 0.944046$

- **Matched using existing approach:**

- **Precision:** $\frac{1794}{1794 + 220} = 0.890765$

- **Recall:** $\frac{1794}{1794 + 896} = 0.666914$

- **F-Measure:** $2 * \left(\frac{0.890765 * 0.666914}{0.890765 + 0.666914} \right) = 0.762755$

	Recall	Precision	F-Measure
NGO:			
<i>DPPMS</i> matched	0.894351	0.99959	0.944046
Matched using existing approach	0.666914	0.890765	0.762755

Table 11. Summary of Recall, Precision and F-Measure for the NGO membership domain.

The NGO membership test results displayed the greatest difference of all three domains. Recall showed a 0.227 gain while the Precision and F-Measure values displayed improvements of 0.109 and 0.181 respectively. The high Precision rate of the DPPMS suggests that pre-processing data is highly beneficial during data matching. A true positive rate increase of 21% was observed.

7.4 Domain result comparison

The metrics of all three domains were compared and relevant inferences made:

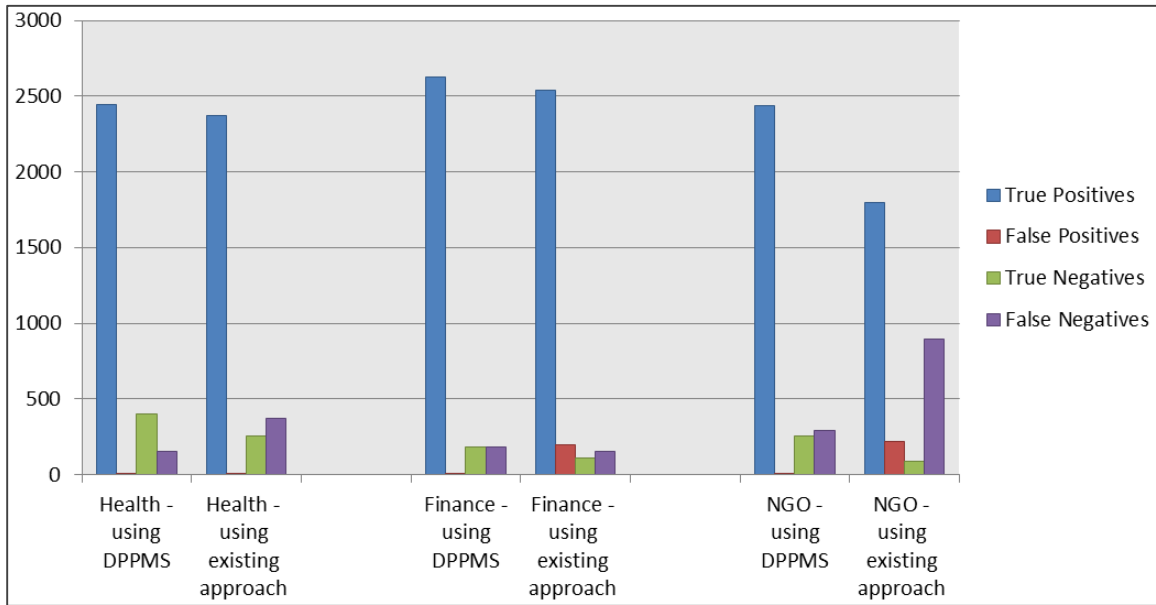


Figure 12. Comparison of result data across the three domains.

The Recall, Precision and F-Measure values for each domain were then calculated:

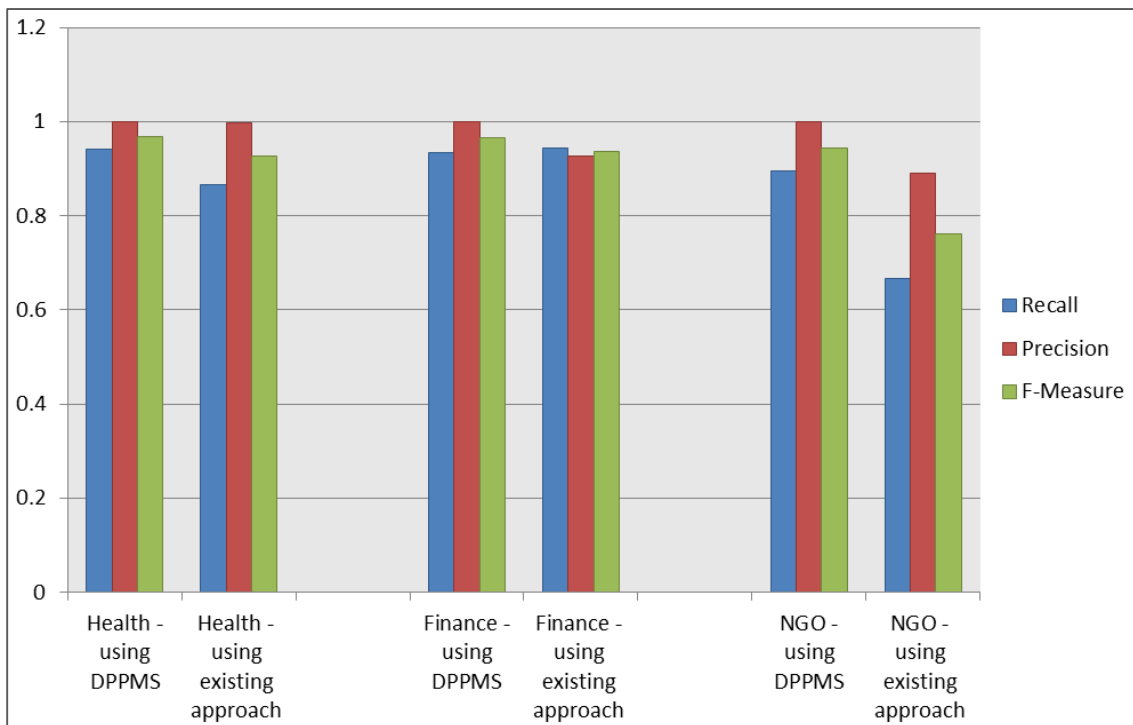


Figure 13. Comparison of the Recall, Precision and F-Measure values.

7.5 Discussion of results

While all three data sets displayed improvements with regard to F-Measure values, the NGO membership data sets displayed the most improvement:

- True Positives increased by 21%
- False Positives decreased by 7%
- True Negatives increased by 5%
- False Negatives decreased by 20%

The increase in true positives and decrease in false negatives suggests that data pre-processing and matching of semi-structured data can lead to an increase in data-match accuracy. The decrease in false positives and increase in true negatives demonstrates that improved data pre-processing will not adversely affect the data-matching process.

Both the financial and health care data sets only displayed an average increase of 3% for their True Positives; this raised the question of why the system was seemingly failing to improve the match quality. Further analysis of these two sectors showed the real problem lay in data that was so badly inaccurate, it was beyond the capability of the DPPMS - only extensive manual intervention or a significantly improved automated system could possibly improve the True Positive match rate. The following contributing factors were noted during the analysis:

- The DPPMS was initially developed based on the requirements of the NGO. The finance and health care domains were included thereafter simply to evaluate the generality of the approach, and in so doing less attention to detail may have resulted in rules being missed. A redesigned system that had either of these data sets as its primary objective may be able to improve the match rate.
- The health and finance sectors have dedicated data entry clerks who are full-time professionals with relevant training and qualifications; as opposed to the NGO volunteers. The majority of data errors were not introduced during data-matching.
- The health and finance sectors have a number of relevant internal data standards and policies that are actively enforced. These are complemented by external legislation and governing bodies that oversee data management practices.
- The data clerks within the health and finance sectors often work on an incentive basis and thus have added motivation to work harder.

While execution speed was not deemed a relevant evaluation goal for this research, it should be noted that the difference in per-record processing time between the current approach and the DPPMS approach was significant. When inputting data manually, the DPPMS took on average 2.3 seconds to process a record. Using the existing matching approach, each record took an average of 83.4 seconds to process. This is the equivalent of a 3600% performance increase. Furthermore, when using the CSV bulk upload functionality, the DPPMS took an average of 0.2 seconds to process each record. Regarding the conundrum first highlighted in Chapter 2, namely the issue of balancing Recall and Precision, the results from all three data sets showed that the DPPMS favoured Precision over Recall. Ultimately, during the design phase the decision was made to sacrifice Recall to improve Precision; this led to a higher false negative rate while allowing the DPPMS to ascertain a high true positive rate and high true negative rate.

7.6 Observations

While the evaluation phase demonstrated that at a system-level, an overall increase in data-match quality was achieved; a number of notable observations were also made at the component level. These are discussed briefly below.

7.6.1 Damerau–Levenshtein distance metric

A major source of contention during the development and evaluation phase was the Damerau–Levenshtein distance metric component. Even after transitioning this functionality from a pure Levenshtein approach to a Damerau–Levenshtein approach, and incorporating a percentage-based analysis step, the results remained unsatisfactory. This component was the root cause of a number of false positives; to mitigate this risk, the default acceptance threshold was lowered to approximately 20%.

For the Church, Circuit and District entities, this threshold was reduced even further to 10%. These reductions rendered the usefulness of this component questionable.

The primary reason for requiring such low thresholds was the extremely close proximity of different entity names, for example: *St Lukes Church of Southern Africa* versus *St Luke*

Church of Southern Africa. In this example, only 1 character separated the 34 character names of the 2 different churches, which equates to a change threshold of less than 3%.

7.6.2 Relocation functionality

The DPPMS was initially designed to automatically move an individual within the hierarchy if the input string had a hierarchy that contrasted with the individual's current position. This proved to be a major source of false positives and a source of frustration to users who would unknowingly move individuals when bad data was provided. To mitigate this risk, a confirmations step was added. This greatly reduced the number of erroneous updates while having a limited impact on the system's overall performance, as it is estimated that less than 5% of all updates are relocation of an individual from one hierarchical position to another.

7.7 Conclusion

Based on the results collected during the evaluation phase, it can be concluded that the gains in data-match accuracy due to data pre-processing justify using such a system. The risks involved are negligible, with all three data sets indicating that data quality is in fact placed at a higher risk when not using expanded data pre-processing.

The following chapter provides a closing summary of this research.

CHAPTER 8. CONCLUSION

8.1 Introduction

This chapter provides an overview of the dissertation as well as the potential for future work.

8.2 Summary

This research focussed primarily on data matching, specifically the challenge of matching low quality and badly structured data to an existing data set.

The driving force behind this research was the day-to-day struggles of a large non-governmental organisation (NGO) with managing the content of their membership database. These included updates arriving in a multitude of different formats, updates that were incomplete, updates that were unstructured, and experts that only had geographically localised knowledge of members and organisational hierarchies. A secondary difficulty was the complex organisational hierarchy which was further compounded by a general lack of data validation processes when updating their database. After a review of the concerns, relevant literature and the current approaches being used manually by data capturers, a system was proposed with the ability to pre-process all incoming data and then match this against the existing data set. An online system was developed, termed the Data Pre-Processing and Matching System (DPPMS). This enabled users to upload database updates which the system would then automatically pre-process, attempt to data match then ultimately update the data set. While initial rule formulation and testing can be a complex process, this is a once off task that if managed effectively, will introduce only limited disruption and overhead to the organisation involved.

After the successful implementation of the DPPMS within the NGO, the system was applied to data sets within the finance and health sectors. With minimal changes to the rule sets and logic, the system was able to cater adequately for the two data sets. While the increase in True Positive match rates was not quite as dramatic as with the NGO data set, an increase in match rates was still observed in both instances.

The development of the DPPMS was challenging at times. The key challenges were the accurate capturing of the rules, determining the Recall/Precision compromise, effectively implementing string distance-metrics and correctly enforcing the organisation's

hierarchical relationships. Through sufficient initial investment in rule development, the reliance on experts in the future is greatly diminished. The conundrum between whether a high Recall or high Precision rate is required is a difficult one and varies from industry to industry; this was demonstrated by the NGO favouring Recall while the finance and health sectors favoured Precision. The difficulty of using distance-metrics to match similar strings was a challenge never fully overcome by this research. To reduce false positive matches with entities that had similar names, match thresholds were reduced to the point whereby their usefulness became questionable. These concerns are potential topics for future research.

The aim of this dissertation was to implement and evaluate a system capable of improving the match rate when importing semi-structured data into a data set. The DPPMS was compared to existing manual methods in the three very different domains of health care, finance and NGO data. The true positive match rate increased by 2.6%, 3% and 21% respectively; Recall by 7.6%, -0.9% and 22.7%; Precision by 0.3%, 7% and 10.9% respectively. Furthermore, no risks or side-effects arose as a result of the DPPMS. It should be noted that even with the DPPMS in place, certain data errors remained unrecoverable. It was unable to match approximately 8% of the provided records; this was largely due to human error during initial data capture. Overall the additional investment in the DPPMS appears worthwhile given its improved handling of semi-structured data updates.

8.3 Future Work

The development of the DPPMS has proven to be somewhat of a catalyst for both the NGO and financial sectors. The NGO is currently investigating the use of a centralised relational database combined with a Web-based user interface, while members of the financial sector are investigating including an expanded pre-processing step within the data validation phase of the KYC data capture process.

The author continues to support both of these endeavours. More specifically, this work has identified a number of related research and development ideas. These are discussed further below.

8.3.1 Quantifying the Recall and Precision compromise

As mentioned in chapter 2, the compromise between Recall and Precision is an accepted challenge within data matching. While this phenomenon is often referred to, limited literature was available on methods to analyse and guide the decision making processes surrounding this conundrum. There is definite scope for further research into a well-structured and soundly-defined approach to handling this compromise.

8.3.2 Data-set level matching

The DPPMS currently performs matching operations at the field and record level; this raised the question of whether the system could be expanded to also perform matching at the data-set level. Government and large corporates may receive a data update where they also need to identify which system or data-set the data update is intended for, e.g. a centralised call centre handling multiple different products and service offerings. It would be feasible to add an intermediate step between data cleaning and the token-level matching phase whereby the DPPMS attempts to identify which data-set is relevant to the update. The high-level matching flow would resemble that shown in figure 14.

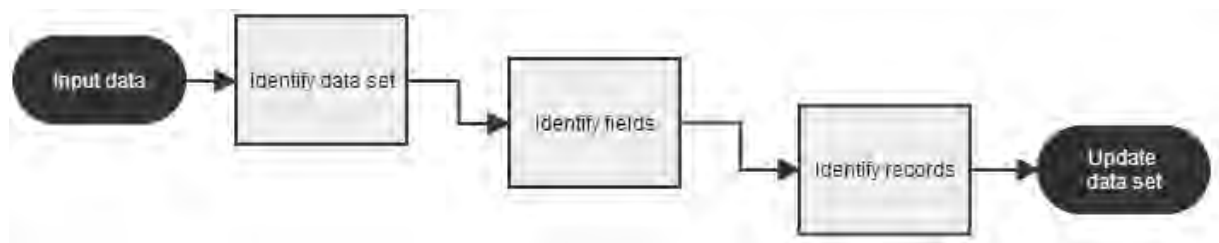


Figure 14. Potential high-level matching flow.

8.3.3 String distance metrics

As discussed in Chapter 7, certain fields in the data set were extremely similar, scoring a distance threshold of less than 3% when analysed using the Damerau–Levenshtein distance metric. With nearly all of the fields being proper nouns, the majority of

dictionary based matching methodologies such as WordNet usage were not applicable. While there is excellent literature available relating to name-matching and record-linkage, there is still a need for further research into technique adaptation in the presence of near-identical target strings.

8.3.4 User interface design

While several interfaces were built as part of the DPPMS implementation, user interaction design is left for future work. This should cover not only interfaces for update submission and confirmation, but also for initial system configuration for new domains. The latter would need to include ways to facilitate the entry of keyword lists, common misspellings, hierarchical and graph relationships, rules, fields and field constraints.

8.3.5 Data mining and data warehousing

This thesis would not be complete without a mention of the potential to apply this research within the data mining field. No other data-related methodology or process relies as heavily on clean and efficiently prepared data than data mining and data warehousing [84]. Improving the data matching process for semi-structured data within a data warehouse can only serve to increase the intrinsic value that data mining is able to leverage from that data warehouse [85, 86].

8.4 Conclusion

This study evolved from a simple implementation of an online database interface into a complex data pre-processing and matching system. In closing, it can be concluded that the use of the DPPMS is valuable when attempting to data match semi-structured data.

REFERENCES

- [1] McKendrick, J. 2012. Today's Data Systems Not quite Ready for Real Time. *Database trends & applications*. 26(3):2-3.
- [2] Deloitte & Touche - South Africa: Firm. 2004. *Roadmap on legal requirements for IDMS implementation by Government Departments*. Pretoria: Pretoria: State Information Technology Agency.
- [3] Khoubati, K., Dwivedi, Y.K., Srivastava, A. & Lal, B. Eds. 2010. *Handbook of Research on Advances in Health Informatics and Electronic Healthcare Applications: Global Adoption and Impact of Information Communication Technologies*. Hershey, PA: Medical Information Science Reference.
- [4] Capocaccia, R., Gatta, G., Roazzi, P., Carrani, E., Santaquilani, M., De Angelis, R., Tavilla, A. & EURO CARE Working Group 2003. The EURO CARE-3 database: methodology of data collection, standardisation, quality control and statistical analysis. *Annals of oncology: Official journal of the European Society for Medical Oncology/ESMO*. 14(Suppl 5):v14-v27.
- [5] SAP Thought Leadership 2012. Making Information Governance a Reality for Your Organization. *Database trends & applications*. 26(3):22-25.
- [6] Massiglia, P., Barker, R. & Veritas. 2002. *The resilient enterprise: recovering information services from disasters*. Mountain View, Calif.: VERITAS Software.
- [7] Hamelin, M. 2012. Cost-effectively dealing with the growing security compliance issue. *Database and network journal*. 42(1):15-16.
- [8] Manji, F. & O'Coill, C. 2002. The missionary position: NGOs and development in Africa. *International affairs*. 78(3):567-583.
- [9] Matheri, M. 2005. Challenges facing the creation of a standard South African address system. *FIG Working Week and 8th Global Spatial Data Infrastructure Conference (GSDI-8)*. 16.
- [10] Shi, W., Fisher, P. & Goodchild, M.F. 2004. *Spatial data quality*. CRC Press.
- [11] Henry, R. 2010. Quiet the perfect cyber crime storm with enterprise threat and risk monitoring. *Database and network journal*. 40(3):14-15.

- [12] Mansuri, I.R. & Sarawagi, S. 2006. Integrating unstructured data into relational databases. *22nd International Conference on Data Engineering - ICDE'06*. 29.
- [13] Pipino, L.L., Lee, Y.W. & Wang, R.Y. 2002. Data quality assessment. *Communications of the ACM*. 45(4):211-218.
- [14] Oliveira, P., Rodrigues, F. & Henriques, P.R. 2005. A Formal Definition of Data Quality Problems. *2005 International Conference on Information Quality (MIT IQ Conference)*.
- [15] Juran, J. & Gryna, F. 1988. *Juran's quality control handbook*. 4th ed. New York: New York: McGraw-Hill.
- [16] Agmon, N. & Ahituv, N. 1987. Assessing data reliability in an information system. *Journal of management information systems*. 4(2):34-44.
- [17] Dasu, T., Vesonder, G.T. & Wright, J.R. 2003. Data quality through knowledge engineering. *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 705.
- [18] Sherman, R. 2004. Seven misconceptions about data quality. *Software world*. 35(6):13-14.
- [19] Redman, T. 2004. Data: An unfolding quality disaster. *DM REVIEW*. 14(8):21-23.
- [20] Wang, Y.R., Strong, D.M. & Guarascio, L.M. 1993. *An empirical investigation of data quality dimensions: A data consumer's perspective*. Total Data Quality Management Research Program, Sloan School of Management, Massachusetts Institute of Technology.
- [21] Abate, M., Diegert, K. & Allen, H. 1998. A Hierarchical Approach to Improving Data Quality. *Data quality*. 4(1):365-369.
- [22] Faculty Development and Instructional Design Center - Northern Illinois University 2012. *Responsible Conduct in Data Management: Data Collection*. Available: http://ori.hhs.gov/education/products/n_illinois_u/datamanagement/dctopic.html [12/20/2013].
- [23] Boskovitz, A., Goré, R. & Hegland, M. 2003. A logical formalisation of the Fellegi-Holt method of data cleaning. In *Advances in Intelligent Data Analysis V*. Springer. 554-565.

- [24] Winkler, W.E. 1994. *Advanced methods for record linkage*. Washington DC: Bureau of the Census.
- [25] Batini, C. & Scannapieca, M. 2006. *Data quality: concepts, methodologies and techniques*. Springer.
- [26] Skoogh, A. & Johansson, B. 2008. A methodology for input data management in discrete event simulation projects. *Proceedings of the 40th Conference on Winter Simulation*. Winter Simulation Conference. 1727.
- [27] Mohamed, H.H., Kheng, T.L., Collin, C. & Lee, O.S. 2011. E-Clean: A Data Cleaning Framework for Patient Data. *2011 First International Conference on Informatics and Computational Intelligence (ICI)*. IEEE. 63.
- [28] Cody, R. 1999. *Data Cleaning 101*. New Jersey: Johnson Medical School.
- [29] Han, J. 2006. *Data mining: concepts and techniques*. Amsterdam: San Francisco, CA: Amsterdam: Elsevier ; San Francisco, CA: Morgan Kaufmann.
- [30] Mayfield, C., Neville, J. & Prabhakar, S. 2009. *A Statistical Method for Integrated Data Cleaning and Imputation*. (09-008). Lafayette: Purdue University: Purdue e-Pubs.
- [31] O'Keefe, R.M. & O'Leary, D.E. 1993. Expert system verification and validation: a survey and tutorial. *Artificial intelligence review*. 7(1):3-42.
- [32] 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. *EconWPA - economics working paper archive*. 510002.
- [33] Blumberg, R. & Atre, S. 2003. The problem with unstructured data. *DM REVIEW*. 13:42-49.
- [34] Abiteboul, S., Buneman, P. & Suciu, D. 2000. *Data on the Web: from relations to semistructured data and XML*. San Francisco: Morgan Kaufmann.
- [35] Levy, A. 1998. *Putting Semi-structured Data to Practice*. Seattle, Washington: University of Washington.
- [36] Yi, X., Allan, J. & Lavrenko, V. 2007. Discovering missing values in semi-structured databases. *Large Scale Semantic Access to Content (Text, Image, Video, and Sound)*. Le Centre De Hautes Etudes Internationales D'informatique Documentaire. 687.

- [37] Good, I.J. 1983. The philosophy of exploratory data analysis. *Philosophy of science*. 50(2):283-295.
- [38] Nobel, N. 2012. Automatic Survey Data Editing: Based on the generalized Fellegi-Holt Paradigm. Master Business Analytics (MBA). VU University of Amsterdam.
- [39] de Waal, T., Pannekoek, J. & Scholtus, S. 2011. *Handbook of statistical data editing and imputation*. Wiley. com.
- [40] Zhang, S., Zhang, C. & Yang, Q. 2003. Data preparation for data mining. *Applied artificial intelligence*. 17(5-6):375-381.
- [41] Fernandez, G. 2002. *Data mining using SAS applications*. Boca Raton, FL: CRC press.
- [42] Herzog, T.N., Scheuren, F.J. & Winkler, W.E. 2007. *Data quality and record linkage techniques*. New York: Springer.
- [43] Sayad, S. 2010. Data Preparation. Toronto: University of Toronto.
- [44] Do, H. & Rahm, E. 2002. COMA: a system for flexible combination of schema matching approaches. *Proceedings of the 28th international conference on Very Large Data Bases*. 610.
- [45] Churches, T., Christen, P., Lim, K. & Zhu, J.X. 2002. Preparation of name and address data for record linkage using hidden Markov models. *BMC medical informatics and decision making*. 2(1):9.
- [46] Witten, I.H. 2011. *Data mining: practical machine learning tools and techniques*. Burlington, MA: Burlington, MA: Morgan Kaufmann.
- [47] Christen, P. 2012. *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection*. Berlin Heidelberg: Springer.
- [48] Moffat, A.A. & Bell, T. 1999. *Managing gigabytes: compressing and indexing documents and images*. Morgan Kaufmann.
- [49] Lee, M.L., Ling, T.W. & Low, W.L. 2000. IntelliClean: a knowledge-based intelligent data cleaner. *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 290.
- [50] Rahm, E. & Do, H.H. 2000. Data cleaning: Problems and current approaches. *IEEE data eng. bull.* 23(4):3-13.

- [51] Jermyn, P., Dixon, M. & Read, B.J. 1999. Preparing clean views of data for data mining. *Twelfth ERCIM Database Research Group Workshop*. 1.
- [52] Drumm, C., Schmitt, M., Do, H. & Rahm, E. 2007. Quickmig: automatic schema matching for data migration projects. *Proceedings of the sixteenth ACM conference on information and knowledge management*. ACM. 107.
- [53] Rahm, E. & Bernstein, P.A. 2001. On matching schemas automatically. *International journal on very large databases (VLDB)*. 10(4):334-350.
- [54] Sullivan, D. 2013. *Data Mining IV: Preparing the Data*. Boston: Boston University.
- [55] The Electoral Commission. 2012. *Data matching schemes to improve accuracy and completeness of the electoral registers – evaluation report*. London: The Electoral Commission.
- [56] Watson, J., Thompson, P. & Scallan, A. 2013. *House of Commons: Political and Constitutional Reform Committee - Oral Evidence - Data-matching pilots for individual electoral registration*. (HC 1109-i). London: The Stationery Office Limited.
- [57] Rahm, E. & Bernstein, P.A. 2001. A survey of approaches to automatic schema matching. *International journal on very large databases (VLDB)*. 10(4):334-350.
- [58] Office for National Statistics. 2003. *National Statistics Code of Practice: Protocol on Data Matching*. London: Office for National Statistics.
- [59] Ives, Z. & Doan, A. 2003. *Data and Schema Matching*. Pennsylvania: University of Pennsylvania.
- [60] Cohen, W. 2003. *Probabilistic Record Linkage: A Short Tutorial*. 1. Pittsburgh: Carnegie Mellon University.
- [61] Madhavan, J., Bernstein, P.A. & Rahm, E. 2001. Generic schema matching with cupid. *International journal on very large databases (VLDB)*. 1(1):49-58.
- [62] Miller, F.P., Vandome, A.F. & McBrewster, J. 2009. *Levenshtein Distance: Information theory, Computer science, String (computer science), String metric, Damerau - Levenshtein distance, Spell checker, Hamming distance*. Orlando: Alpha Press.
- [63] Damerau, F.J. 1964. A technique for computer detection and correction of spelling errors. *Communications of the ACM*. 7(3):171-176.

- [64] Bard, G.V. 2007. Spelling-error tolerant, order-independent pass-phrases via the Damerau-Levenshtein string-edit distance metric. *Proceedings of the fifth Australasian symposium on ACSW frontiers-Volume 68*. Australian Computer Society, Inc. 117.
- [65] Kirani, S., Zualkernan, I.A. & Tsai, W. 1992. Comparative evaluation of expert system testing methods. *Proceedings of Fourth International Conference on Tools with Artificial Intelligence - TAI '92*. 334.
- [66] Yatskevich, M. 2003. *Preliminary evaluation of schema matching systems*. (DIT-03-028). Trento, Italy: University of Trento.
- [67] Christen, P. & Goiser, K. 2007. Quality and complexity measures for data linkage and deduplication. In *Quality Measures in Data Mining*. Springer. 127-151.
- [68] Giusti, A. & Ritter, G. 2013. *Classification and data mining*. Berlin: Berlin: Springer Verlag.
- [69] Pang-Ning, T., Steinbach, M. & Kumar, V. 2006. Classification: basic concepts, decision trees and model evaluation. *Introduction to data mining*. 1:145-205.
- [70] Maimon, O.Z. & Rokach, L. 2005. *Data mining and knowledge discovery handbook*. Springer.
- [71] Peukert, E., Eberius, J. & Rahm, E. 2012. A self-configuring schema matching system. *IEEE 28th International Conference on Data Engineering (ICDE)*. IEEE. 306.
- [72] Nishida, F., Takamatsu, S., Tani, T. & Doi, T. 1988. Feedback of correcting information in postediting to a machine translation system. *Proceedings of the 12th conference on Computational linguistics (COLING)*. Association for Computational Linguistics. 476.
- [73] Nadig, S. & Murthy, A. 2013. *Anti-money laundering principles for alternate payment methods*. Bangalore, India: Infosys Ltd.
- [74] Clarke, A. 1992. Researching Health Care: Designs, Dilemmas, Disciplines. *Critical public health*. 3:4-41.
- [75] Tsumoto, S. & Tanaka, H. 1996. Automated Discovery of Medical Expert System Rules from Clinical Databases Based on Rough Sets. *Proceedings of KDD 96 - Association for the Advancement of Artificial Intelligence*. 63.

- [76] Braa, J. & Hedberg, C. 2002. The struggle for district-based health information systems in South Africa. *The information society*. 18(2):113-127.
- [77] Terrell, T. 1999. The end of money and the struggle for financial privacy. *Quarterly journal of austrian economics*. 2(2):87-91.
- [78] Donovan, K. 2012. Mobile money for financial inclusion. *Information and communication for development*.:61-74.
- [79] Jenkins, B. 2008. *Developing mobile money ecosystems*. Washington, DC: International Finance Corporation and Harvard Kennedy School.
- [80] Hu, W.C., Lee, C. & Kou, W. 2005. *Advances in security and payment methods for mobile commerce*. IGI Global.
- [81] Dilley, D.K. 2008. *Essentials of banking*. Hoboken, NJ: John Wiley and Sons.
- [82] De Koker, L. 2004. Client identification and money laundering control: perspectives on the Financial Intelligence Centre Act 38 of 2001. *Journal of south african law*. 4:715-746.
- [83] Jürjens, J., Fernandez, E.B., France, R.B., Rumpe, B. & Heitmeyer, C. 2005. Critical systems development using modelling languages (CSDUML'04): current developments and future challenges (report on the third international workshop). In *UML Modelling Languages and Applications*. Springer. 76-84.
- [84] Hand, D.J., Mannila, H. & Smyth, P. 2001. *Principles of data mining: adaptive computation and machine learning*. Cambridge, Massachusetts: The MIT Press.
- [85] Bhowmick, S.S. 2004. *Web data management: a warehouse approach*. New York: New York: Springer.
- [86] Devlin, B. & Cote, L.D. 1996. *Data warehouse: from architecture to implementation*. Addison-Wesley Longman Publishing Co., Inc.

APPENDIX

NGO database schema

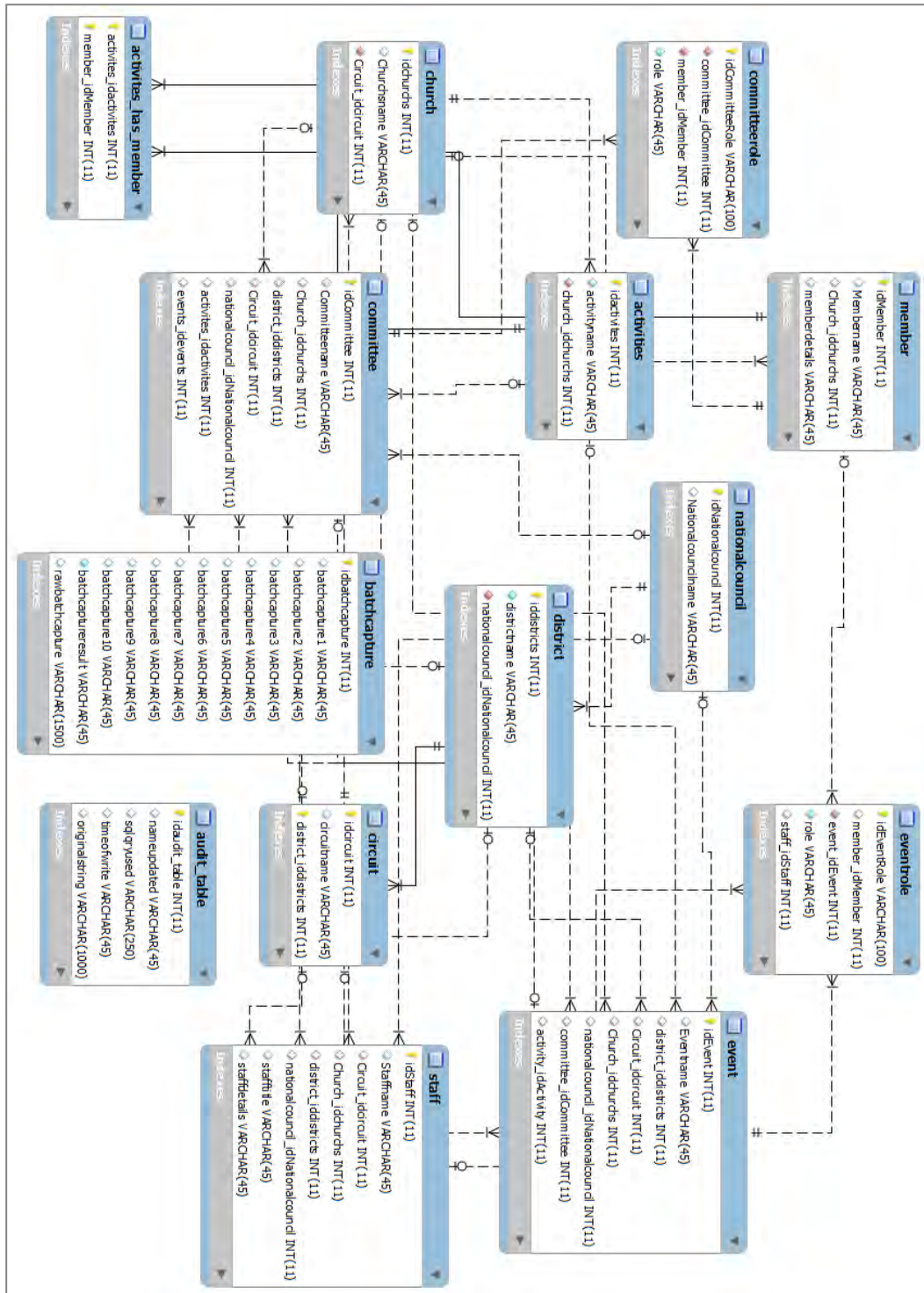


Figure 15. NGO membership database schema.