A Framework for Providing Integrated Strategic Information for the Management of the Antiretroviral Treatment Program in the Free State, South Africa

Thesis submitted by

Jacobus Eduan Kotzé

Student number: 1993261971

to the

Department of Computer Science and Informatics
Faculty of Natural and Agricultural Sciences
University of the Free State, South Africa

in the fulfilment of the requirements for the degree

PHILOSOPHIAE DOCTOR (Computer Information Systems)

November 2008

Promoter: Prof. T. McDonald

GLOSSARY

AIDS	Acquired Immunity Deficiency Syndrome
ART	Antiretroviral Therapy
ARV	Antiretroviral
(The) Author	The author of this thesis
BI	Business Intelligence
CD4	Cluster of Differentiation 4. CD4 cells are a type of white blood cell and are an important part of the immune system of a human.
CDC	Centre for Disease Control, United States of America
CIS	Clinic Information System
CPR	Computer-based Patient Record
DDL	Data Definition Language
DML	Data Manipulation Language
DSS	Decision Support System
DW	Data Warehouse
EMR	Electronic Medical Record
EPR	Electronic Patient Record
FSDOH	Free State Department of Health
GRLS	General Record Linkage System
HIV	Human Immunodeficiency Virus
HIS	Hospital Information System
HOD	Head of Department or also called Chief Executive Officer
LPR	Longitudinal Patient Record
MOLAP	Multidimensional Online Analytical Processing
MPM	Meditech's Medical Practice Management Suite
MRC	Medical Research Council, South Africa
MRI	Meditech's Medical Record Interface
NDOH	National Department of Health, South Africa
NHC/MIS	National Health Care Management Information System of South Africa
NHIS/SA	National Health Information System of South Africa
OLTP	Online Transaction Processing
OLAP	Online Analytical Processing
PDA	Personal Digital Assistant
PE	Patient Evaluation (or also known as Clinical Investigations)
PHC	Primary Health Care

SITA	State Information Technology Agency, South Africa
SQL	Structured Query Language
PL/SQL	Procedural Language/Structured Query Language is Oracle Corporation's proprietary procedural extension to the **SQL** database language
SAN	Storage Area Network
STD	Sexually Transmitted Disease
TB	Tuberculosis is primarily an illness of the respiratory system, and is spread by coughing and sneezing.
UCT	University of Cape Town, South Africa
VCT	Voluntary Counseling and Testing
VL	Viral Load of the HIV virus
WAN	Wide Area Network
WOLAP	Web-Based Online Analytical Processing
WHO	World Health Organization

ACKNOWLEDGEMENTS

The writing of this thesis together with constructing the data warehouse has been the most challenging and rewarding experience of my life. It has, however, been made possible by the contributions of many people. My deepest thanks go to all of them. In particular, I would like to thank the following persons:

1. My promoter, Prof. Theo McDonald, for his encouragements, comments, advice and mentorship during the writing of this thesis. He gave a lot of his valuable time in teaching me how to write. I will surely miss our fruitful discussions on data warehousing and probabilistic matching.

2. Dr. Ronald Chapman, who granted me a study-on-instruction to complete a PhD, sponsoring the FSDOH data warehouse project, allowing me to attend The Data Warehouse Institute (TDWI) World Conference in San Diego and also to present a paper at IRMA in Vancouver, Canada.

3. Mr. Bennie de Winnaar, my immediate supervisor and manager, for all his support and mentoring during the construction of the data warehouse. His generous leadership style allowed me to work independently, explore different scenarios and to freely experiment in finding the best possible solution with the limited resources that was available.

4. Mr. Andre Venter (Meditech), for all the development work he has done to customize the MPM application to the needs of construction the data warehouse project.

5. Mrs. Susan Robertson (MPM Implementation Team Leader), Mrs. Wendy Adolph and Miss. Jonel Jonker (MPM Implementation Team) for all their dedication and hard work to implement the Meditech MPM application across the Free State.

6. Mr. Stanley Coetzer, who developed the LPR linkage web application.

7. Dr. Lara Fairral (Knowledge Translation Unit, University of Cape Town Lung Institute, University of Cape Town) for sponsoring my trip to the University of Bern, Switzerland and the opportunity to learn GRLS.

8. Mr Adrian Spörri-Fahrni and Mr Kurt Schmidlin (Institute of Social and Preventive Medicine, University of Bern, Switzerland) for their patience and using their time in teaching me how to use GRLS.

9. Dr Matthias Egger (Institute of Social and Preventive Medicine, University of Bern, Switzerland), for making a GRLS server available at no cost for all the probabilistic linkage experiments.

10. Brenda Woodbridge, General Manager TDWI, for awarding me a TWDI scholarship to attend the TWDI World Conference in San Diego. This training formed one of the cornerstones in my theoretical knowledge into the field of data warehousing.

11. Most importantly, my wife Yolandi, for all her patience, love, encouragement during the writing of this thesis. I also received wonderful support from my parents Jaco and Juliet as well as my wife's parents, Jan and Marinda, for which I am very grateful.

12. My Father in Heaven who gave me the strength and perseverance to complete this data warehousing project and thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES (continue)

LIST OF FIGURES (continue)

LIST OF TABLES

LIST OF TABLES (continue)

CHAPTER 1 - INTRODUCTION

1.1. Introduction

The Acquired Immune Deficiency Syndrome (AIDS) epidemic, caused by the Human Immunodeficiency Virus (HIV) is a global crisis which threatens development gains, economies, and societies. The epidemic has evolved in different ways in different parts of the world, and at varying speeds. In many regions it is still in its early stages. AIDS is unique in human history in its rapid spread, its extent and the depth of its impact. In its yearly Global Report on AIDS, the United Nations reported that in the 20 years of the disease's existence, almost 20 million people are dead and 33 million people (range: 30.3 – 36.1 million) worldwide are living with HIV (UNAIDS, 2008c:32).

Within Sub-Saharan Africa, where the epidemic began the earliest and the HIV prevalence is the highest, African countries have death rates not seen since the 1950's or 1960's. According to UNAIDS (2008a:5), Sub-Saharan Africa remains the most heavily affected by HIV, accounting for 67% of all people living with HIV and for 72% of AIDS deaths in 2007.

In South Africa the epidemic has a devastating impact which creates profound suffering on individuals and their families, and the impact on the socio-economic level is of great concern. HIV/AIDS is a major threat to the most productive segment of the labour force and contributes to reduced earnings, imposing huge costs on enterprises in all sectors through decreasing productivity, increasing labour costs, and loss of skills and experience. HIV/AIDS is affecting fundamental rights at work, with respect to workers and people living with and affected by HIV/AIDS. The epidemic and its impact strike hardest at vulnerable groups including women and children, therefore increasing existing gender inequalities and exacerbating the problem of child labour (UNAIDS, 2004). The HIV/AIDS epidemic threatens the viability of health-care systems. Treating AIDS and related opportunistic infections are placing heavy burdens on the health-care system of South Africa and throughout the world.

The eradication of HIV/AIDS represents one of humanity's greatest challenges, which requires co-operation, and comprehensive collaboration between science, governments, social institutions, the media, the professions, and the general public. **In this endeavour strategic information plays a major role.**

1.2. HIV/AIDS and ARV Treatment

South Africa's Health Environment faces a challenging situation in trying to deal with the HIV/AIDS epidemic. The UNAIDS Global Report estimated that the number of AIDS related deaths in South Africa in 2007 ranged anywhere between 270 000 and 420 000 (UNAIDS, 2008b:217).

Currently the 15-49-year-olds carries the highest infection of HIV/AIDS in South Africa. Dramatic rises in the number of orphans are expected as the disease dissipates families and kills of one of the parents or in most cases, both parents. One must also take note that the 15-49-year-olds are mostly the income-generators of a family which in turn means fewer taxes to support the country and the social welfare system.

In response to this epidemic the South African Government created the HIV/AIDS and Sexually Transmitted Disease (STD) Strategic Plan. The purpose of the plan is to provide a broad national framework around four priority areas: prevention; treatment, care and support; research, monitoring and evaluation; human and legal rights. In November 2003, after considerable sustained pressure from advocacy groups, the government adopted the Operational Plan for Comprehensive HIV and AIDS Treatment and Care, which included the provision of Antiretroviral Therapy (ART) in the public health sector.

1.3. Problem statement

The roll-out of the ART programme was proving to be a slow process in South Africa and it was not any different in the Free State (AIDS Foundation, 2006). A patient information system was deployed by the Province to supplement the rollout process by gathering data and providing all the basic patient antiretroviral information.

The lack of strategic information was prominent if one takes a closer look at the Free State antiretroviral treatment programme rollout and supporting patient information system. The patient information system was a traditional online clinical system, dealing with the bread-and-butter issues of accumulating data on a patient. Very little functionality was provided to deal with the complexities of managing the clinical outcomes of the ART programme. To add to the problem, other operational systems had to be interrogated to gain an understanding of the impact the rollout of ARVs had. These operational systems ranged from standalone Human Resource systems to information systems accumulating data on tuberculosis which is closely related to HIV/AIDS. No mechanism or platform existed to provide management with integrated strategic information to manage the business process intelligently. In an

attempt to overcome this lack of strategic information, this study will focus on the challenges and solutions to overcome this shortfall. The following section will describe the methodology and objectives required to overcome the information challenges that the ART programme presented.

1.4. Research Methodology

The research methodology followed by this study will be action research. Fennessy and Burnstein (2000) quotes Baskerville and Wood-Harper (1998) by defining action research as "*a cognitive process that depends on social interaction between the observers and those in their surroundings*". Butler, Feller, Pope, Murphy and Emerson (2006) noted that in action research projects, researchers collaborate with practitioners to solve practical problems while expanding scientific knowledge.

Baskerville (1999) cites Blum (1955) and argues that action research can be described by a simple two stage process. During the *diagnostic stage,* a collaborative analysis of the social situation is performed by the researcher and the subjects. The diagnostic phase is followed by the *therapeutic stage* that involves experimentation. In this stage changes are introduced and the effects are studied.

A more precise definition of action research can be drawn from the work done by Baskerville (1999) where the author characterizes information system action research as follows:

1. "Action research aims at an increased understanding of an immediate social situation, with emphasis on the complex and multivariate nature of this social setting in the information systems (IS) domain.
2. Action research simultaneously assists in practical problem solving and expands scientific knowledge. This goal extends into two important process characteristics: First, there are highly interpretive assumptions being made about observation; second, the researcher intervenes in the problem setting.
3. Action research is performed collaboratively and enhances competencies of the respective actors. A process of participatory observation is implied by this goal. Enhanced competencies (an inevitable result of collaboration) are relative to the previous competencies of the researchers and subjects, and the degree to which this is and its balance between the actors will depend upon the setting.
4. Action research is primarily applicable for the understanding of change processes in social systems."

The action research description (Susman and Evered, 1978) details a five phase, cyclical process. The approach first requires the establishment of a client-system infrastructure or research environment. Then, five identifiable phases are iterated: 1) diagnosing, (2) action planning, (3) action taking, (4) evaluating and (5) specifying learning

Baskerville (1999) illustrates this action research structural cycle (Figure 1-1) and also provides an explanation of these components.

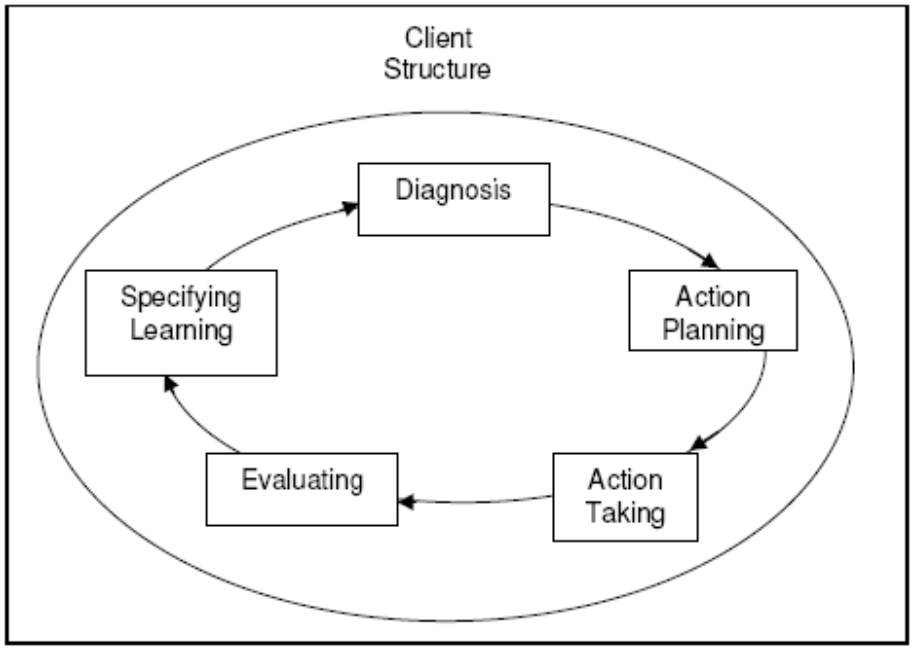


Figure 1-1: The action research cycle (Baskerville, 1999)

The *client-system infrastructure* is the specification and agreement that constitutes the research environment and provides the conditions under which action and change may be specified. The infrastructure should also define the responsibilities of the client and the researcher to each other in a collaborative nature of undertaking. By referring to this explanation it is important to point out that the client will be the FSDOH and the researcher will be the author of this thesis. The research environment will be the Free State Province and will include staff working for the FSDOH at different levels of management. Outside entities will also be involved in the evaluation of the data warehouse solution, due to their relationship in the study on the effects of antiretroviral drugs on combating the spread of the disease.

Diagnosing corresponds to the identification of the primary problems that are causing the organization's desire for change. The diagnoses will develop theoretical assumptions about the nature of the problem domain that needs solving.

Action planning is the collaborative effort of the researcher and the client to identify organizational actions to relieve or improve the specified problems. The output is a plan that establishes the target of change and the approach to change.

Action taking then implements the planned action in a collaborative manner between the researcher and the client.

A collaborative *evaluation* of the implemented plan is done to determine if the changes had the desired outcome. This includes determining whether the effects of the action were realized, and whether the problems have been relieved. Where the change was successful, the evaluation should indicate whether the actions undertaken were the sole cause of success. In the case of where the action was unsuccessful, the reasons should be identified and the action plan for the next iteration needs to be established.

Specifying learning is formally undertaken last, but is usually an ongoing activity. The organizational norms should be restructured to reflect the new knowledge gained during the research. Where the change was unsuccessful, additional knowledge should be added in preparation for the next action research cycle. Where the change was successful, the actions involved should be documented to aid future research.

Baskerville (1999) emphasizes the point that action research produces highly relevant research results, because it is grounded in practical action, aimed at solving an immediate problem while carefully informing theory. Due to the fact that the author of this thesis will be *actively involved* in constructing a data warehouse and not only an observer, the action research methodology fits perfectly.

1.5. Objective of research study

The main objective of this study is to supply comprehensive integrated strategic information for the management of the ART programme in the Free State Department of Health (FSDOH). The main objective will be reached by means of a phased approach. For each phase a sub-objective will be formulated. The action research methodology will be applied to each phase. The following sub-objectives need to be achieved:

Phase 1: Supply strategic ART information from individual data marts.

- Diagnosis
    - Gain an understanding of the ART model of care adopted by FSDOH.
    - Gain an understanding of the current problems with the ART information systems.
- Action planning
    - Gain an understanding of current data warehouse principles and technologies.
    - Deploy the necessary data warehouse infrastructure.
- Action taking
    - Design data warehouse architecture.
    - Develop individual data marts corresponding to the business processes in need of change.
    - Develop a business intelligence solution that provides OLAP and ad-hoc query capabilities.
- Evaluation
    - Do a usability study to determine the effectiveness, efficiency and satisfaction of the data warehousing solution for management.
- Specifying learning
    - Document what has been learned by Phase 1.

Phase 2: Supply comprehensive integrated strategic ART information from a longitudinal patient record which is constructed by linking the individual data marts.

- Diagnosis
    - Identify the shortcoming and problems of Phase 1.
    - Understand the general principles of record linkage.
- Action planning
    - Formulate a plan to link the individual data marts.
- Action taking
    - Link the individual data marts by applying record linkage principles.

    - Develop a business intelligence solution.
- Evaluation
    - Do a usability study to determine the effectiveness, efficiency and satisfaction of the data warehousing solution for management.
- Specifying learning
    - Document what has been learned by Phase 2.

1.6. Hypothesis of the study

The following specific research hypothesis is proposed:

A framework for delivering comprehensive integrated strategic information for the management of the rollout of antiretroviral treatment in the Free State Department of Health can be successfully implemented.

1.7. Chapter Outline

The chapter outline of the study will follow the cyclical approach of the action research methodology that was proposed earlier.

For Phase 1, the *problem diagnosis* phase will be covered in Chapter 2 and 3. *Action planning* will be covered in Chapter 4 and 5. The *action taken* phase will be covered in Chapter 6 and 7. The *evaluation* and *specify learning* phases will be covered in Chapter 8. Listed below are brief descriptions on each of the chapters mentioned.

Chapter 2 provides an overview of HIV and AIDS and also discusses the use of antiretroviral treatment as a possible method to curb the impact of the disease.

Chapter 3 provides a comprehensive overview on the existing Information Systems in use at the Free State Department of Health to support the ART Programme. This chapter will conclude with shortfalls within the existing Information System.

Chapter 4 will provide a motivation for selecting a data warehouse architecture and design methodology for the FSDOH. This chapter will conclude with a literature study on work done in the Healthcare field and then compare the findings with the proposed content of this thesis.

Chapter 5 provides insight to data warehousing features that will be used to maximize the flexibility and performance of the FSDOH data warehouse architecture.

Chapter 6 provides in detail the construction phase of the data warehouse and all the data marts to provide strategic information for the FSDOH.

Chapter 7 will elaborate on the development of a business intelligence solution that will provide the analytical capabilities to users of the FSDOH data warehouse.

Chapter 8 will evaluate the managerial outcomes of the business intelligence and data warehouse solution. Shortfalls and possible solutions will be identified which in turn will contribute to new knowledge and learning.

For Phase 2, the *problem diagnosis* phase will be covered in Chapter 9. *Action planning* will be covered in Chapter 10. The *action taken* phase will be covered in Chapter 11 and partially in Chapter 12. The *evaluation* and *specify learning* phases will be covered at the end of Chapter 12.

Chapter 9 provides in detail the construction of additional data marts for the data warehouse which were identified from the *evaluation* and *specify learning* phase of Phase 1.

Chapter 10 will outline all the different record linkage mechanisms and will provide the theoretical background for constructing a single data warehouse using independent data marts. This chapter will conclude with a literature study on work done in the Healthcare field and then compare the findings with the proposed content of this thesis.

Chapter 11 will elaborate on linking the independent data marts using probabilistic record linkage mechanisms. The chapter will also outline the probabilistic findings that were discovered when using the probabilistic matching tool (GRLS).

Chapter 12 provides in detail the construction of a longitudinal patient record that will provide integrated strategic information for the management of the ART programme.

Chapter 13 will conclude the study by summarizing the main findings of this thesis and highlighting the contribution of this research to new knowledge.

1.8. Chapter Summary

This chapter introduced the worldwide problem of HIV and AIDS and the effect it has on South Africa today. In response to this epidemic the South African Government created the HIV/AIDS and Sexually Transmitted Disease (STD) Strategic Plan which includes the rollout of antiretroviral therapy programme. The research problem of addressing the challenges in providing strategic information to manage the implementation an antiretroviral therapy programme in the Free State was described. Several research objectives were proposed in an effort to solve these challenges. The following chapter will provide a deeper understanding of the HIV and AIDS status. The Free State model of ART care will also be examined to provide an overview of the methodology followed by the Free State Department of Health on providing patients with the drugs.

CHAPTER 2 - HIV/AIDS, ANTIRETROVIRAL TREATMENT AND THE MODEL OF CARE IN THE FREE STATE

2.1. Introduction

The previous chapter introduced the worldwide problem of HIV and AIDS and the effect it has on South Africa today. The South African Government in response to this disease proposed the implementation of a national antiretroviral therapy programme in the Public Health sector. The research problem of addressing the challenges in providing strategic information to manage the implementation an antiretroviral therapy programme in the Free State was described. Several research objectives were proposed in an effort to solve these challenges.

This chapter will provide a detailed theoretical discussion on HIV and AIDS. The usage of antiretroviral drugs to combat the impact of the disease will also be examined. The chapter will conclude with an outline on the ART model of care, which was adopted by the FSDOH. In terms of the action research methodology, this chapter will form the cornerstone of the *problem diagnosis* phase.

2.2. HIV Status

HIV/AIDS first became public notice in 1981 (UNAIDS, 2004). The potentially fatal virus is found in all regions of the world. It is not restricted to race, sexual orientation, affected nations, nor is it affected by political or ideological stances or cultural values. According to the Canadian Centre of Occupational Health and Safety (CCOHS, 1997) the HIV virus weakens the body's immune system and causes the AIDS disease and presently there is ***no cure***. The full name of AIDS - *Acquired Immune Deficiency Syndrome* - describes several of the characteristics of the diseases as follows: *Acquired* indicates that it is not an inherited condition; *Immune deficiency* indicates that the body's immune system breaks down resulting in the fact that a person with HIV becomes vulnerable to a range of opportunistic infections which normally the body could fight off. It is one or more of these infections which ultimately cause death. *Syndrom*e indicates that the disease results in a variety of health problems (CCOHS, 1997).

HIV is transmitted through body fluids of infected persons such as blood, blood products, semen, vaginal secretions, and mother to child transmission during birth or through breastfeeding. HIV is not transmitted by casual physical contact, mosquito or insect bites, kissing, coughing or sneezing, sharing toilets or washing facilities, consuming food or drink handled by someone who has HIV.

2.2.1. HIV and AIDS Globally

The annual number of new HIV infections declined from 3.0 million in 2001 to 2.7 million in 2007 (UNAIDS, 2008a:5). An alarming statistic is the fact that young people aged 15-24 account for an estimated 45% of new HIV infections worldwide (UNAIDS, 2008c:33). Figure 2-1 depicts the global picture of HIV infection in 2007.

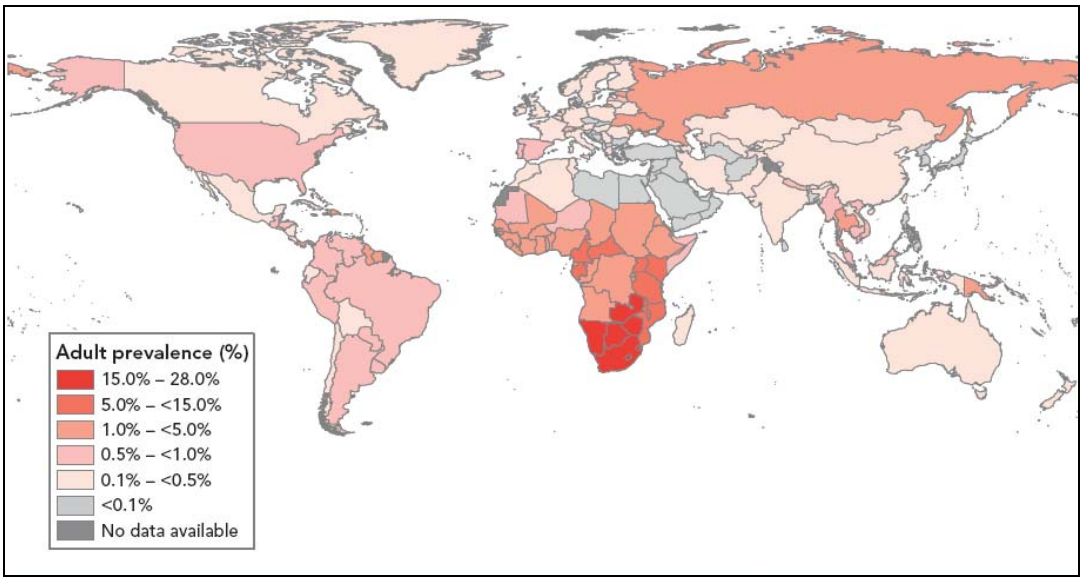


Figure 2-1: A global view of HIV infection, 2007 (UNAIDS, 2008c:33)

2.2.2. HIV and AIDS in Sub-Saharan Africa

Sub-Saharan Africa has just over 10% of the world's population, but is home to 67% of all people living with HIV worldwide —some 22 million (range: 20.5 – 23.6 million) (UNAIDS, 2008b:214). According to (UNAIDS, 2008c:39) almost 75% of all AIDS related deaths occurred in Sub-Saharan Africa in 2007.

In 2007 alone, an estimated 1.9 million people (range: 1.6–2.1 million) in the Sub-Saharan Africa region became newly infected (UNAIDS, 2008c:39), while 1.5 million (range: 1.3 – 1.7 million) died of AIDS (UNAIDS, 2008b:217). Among young people (15–24 years of age), 3.2% of women (range: 2.6 – 3.8%) and 1.1% of men (range: 0.8 – 1.4%) were living with HIV by the end of 2007 (UNAIDS, 2008b:217).

Southern Africa continues to bear a disproportionate share of the global burden of HIV: 35% of HIV infections and 38% of AIDS deaths in 2007 occurred in that sub region (UNAIDS, 2008c:32).

A combination of factors seem to be responsible for this, including: poverty and social instability; high levels of sexually transmitted infections; the low status of women; sexual violence; high mobility (particularly migrant labour); and lack of leadership (AIDS Foundation, 2006).

HIV's impact on adult mortality is greatest on people in their twenties and thirties, and is proportionately larger for women than men. In low- and middle-income countries, mortality rates for 15–49-year-olds living with HIV are now up to 20 times greater than death rates for people living with HIV in industrialized countries. This reflects the stark differences in access to antiretroviral therapy. In low- and middle-income countries, mortality generally varies between two and five deaths per 1000 person years (PY) for people in their teens and twenties. However, HIV-infected individuals in these age groups experience death rates of 25–120 per 1000 PY, rising to 90–200 per 1000 PY for people in their forties (UNAIDS, 2004).

Until recently, low- and middle-income countries had extended life expectancy significantly. However, since 1999, primarily as a result of AIDS, average life expectancy has declined in 38 countries. In seven African countries where HIV prevalence exceeds 20%, the average life expectancy of a person born between 1995 and 2000 is now 49 years—13 years less than in the absence of AIDS (see figure 2-2). In Swaziland, Zambia and Zimbabwe, the average life expectancy of people born over the next decade is projected to drop below 35 years in the absence of antiretroviral treatment (UNAIDS, 2004)

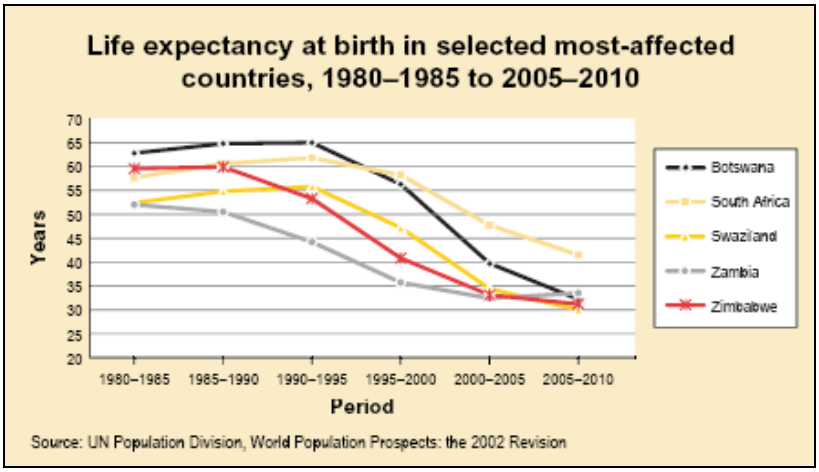


Figure 2-2: Life Expectancy at Birth (UNAIDS, 2004:42)

2.2.3. HIV and AIDS in South Africa

Given the numbers of people infected and dying, South Africa is regarded as having the most severe HIV epidemic in the world. Figure 2-3 depicts the population size of South Africa, with and without AIDS and provides a hypothetical population size in 2025 to illustrate the impact of the disease.

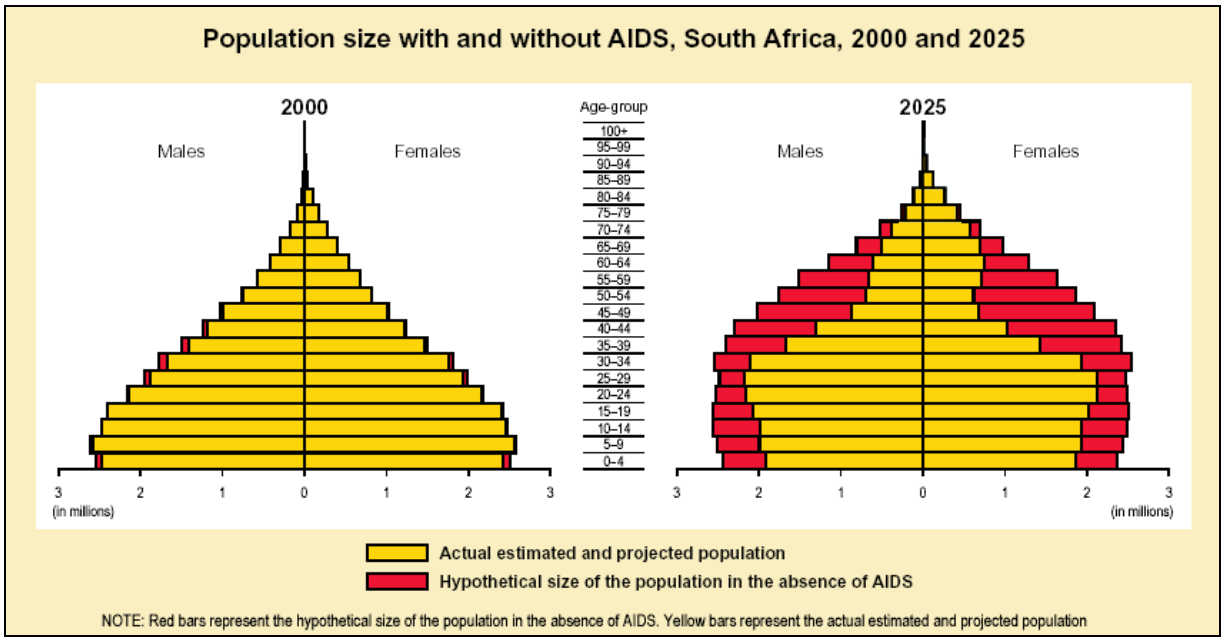


Figure 2-3: Population size with and without AIDS, South Africa (UNAIDS, 2004:42)

This epidemic is still seven years away from peaking in terms of the numbers of projected AIDS related deaths (AIDS Foundation, 2006). Just over 5.7 million people (UNAIDS, 2008b:214) out of a total of 47.85 million (STATSSA, 2007) South Africans were HIV positive in 2007, giving a total population prevalence rate of 11.9%.

According to Dorrington, Johnston, Bradshaw and Danel (2006) the number of people infected with HIV is beginning to stabilise at around 6 million people. This is because the number of new infections has declined to the point where it nearly matches the number of people dying from AIDS. The picture (see figure 2-4) below shows the waves of the epidemic according to the default scenario of ASSA2003 (Dorrington et al., 2006). ASSA2003 is the latest demographic and AIDS model that was developed by the Actuarial Society of South Africa (ASSA) and uses data from several sources to project the potential course of the epidemic and the demographic impact that it is having.

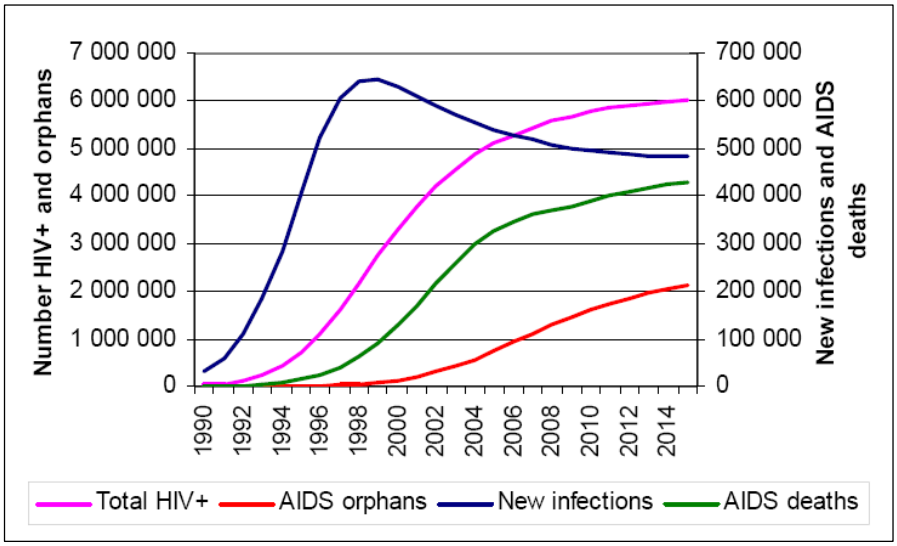


Figure 2-4: The Waves of the AIDS Epidemic (Dorrington et al., 2006:3)

According to the National Department of Health (2007:17), the annual HIV and Syphilis Antenatal Sero-Prevalence Survey report presented the fact that the national average proportion of HIV positive women attending antenatal clinics in 2007 to be 28.0%. The province of KwaZulu-Natal continues to have the highest prevalence, at 37.4%.

2.2.4. HIV and AIDS in the Free State

The Free State province is one of the smallest of the nine provinces in South Africa (2.857 million people) and has the second highest prevalence among provinces in South Africa after Kwazulu-Natal (KZN). The latest HIV prevalence rate among pregnant women was reported to be at 33.5% in 2007 (National Department of Health, 2007).

According to Chapman (2003) using the ASSA 2000 Model, it is estimated that in the Free State:

- Approximately 480 000 people are HIV positive;
- Approximately seven percent (7%) of all HIV infected patients are in WHO Stage 4 AIDS defining illness, which is approximately 33 600 patients;
- Annually, 28 290 patients will develop WHO Stage 4 AIDS defining illness.

An alarming statistic according to the 2004 midyear statistics is the fact that 85.2% of the Free State population has no medical insurance (Free State Department of Health, 2006). This uninsured population

(2,434,606) will therefore be mainly dependent on public health services for the provisioning of antiretroviral drugs in the future, should they be exposed to the HIV virus.

Tabled below is an outline of the insured and uninsured population figures (see table 2-1) per district which is followed by a map of the Free State. The map indicates the major towns and cities where ARV treatment must be made available (see figure 2-5).

Table 2-1: Population Statistics (Free State Department of Health, 2006).

Health District	Population	Insured	Uninsured
Xhariep	132,070	19,546	112,524
Motheo	736,292	108,971	627,321
Lejweleputswa	762,858	112,903	649,955
Thabo Mofutsanyana	738,328	109,273	629,055
Northern Free State	487,971	72,220	415,751
Province	**2,857,519**	**422,913**	**2,434,606**

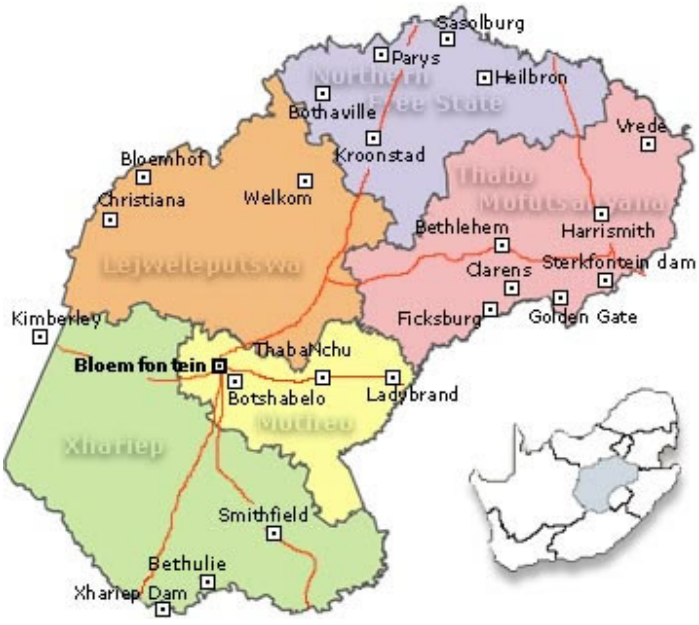


Figure 2-5: Map of the Free State Province

2.3. HIV Antiretroviral Drug Treatment

Antiretroviral therapy (ART) is the main type of treatment for HIV or AIDS. It must be remembered that this is not a cure, but it can prolong a person's life with several years if the drugs are taken every day. It will also stop a person from becoming ill for many years and add to that person's productivity in society.

HIV is a virus and when it is in a cell in the body it produces new copies of itself. With these new copies, HIV can go and infect other previously healthy cells. ART for HIV infection consists of drugs which work by slowing down the reproduction of the HIV virus in the body. The drugs are often referred as antiretrovirals, anti-HIV drugs or HIV antiviral drugs (Kanabus, 2006).

The World Health Organization (WHO) recommends that before anyone starts treatment, a basic clinical assessment should be carried out. This should include: documentation of past medical history, identification of current and past HIV-related illnesses, identification of other medical conditions that might influence the timing and choice of ART, and current symptoms and physical signs of other medical conditions such as TB or pregnancy (Kanabus, 2006).

Once this assessment has been carried out, it will be known which stage of HIV disease the person has. The WHO has a method of describing people with HIV as being at different stages of HIV infection, according to the different clinical symptoms they may have. This is known as the WHO staging system for HIV infection which is not dependent on testing (Kanabus, 2006).

WHO Clinical Stage I:

- Asymptomatic
- Generalized lymphadenopathy

Performance scale 1: asymptomatic, normal activity

WHO Clinical Stage II:

- Weight loss <10% of body weight
- Minor mucocutaneous manifestations (seborrheic dermatitis, prurigo, fungal nail infections, recurrent oral ulcerations, angular cheilitis)
- Herpes zoster within the last five years
- Recurrent upper respiratory tract infections (i.e. bacterial sinusitis)

And/or performance scale 2: symptomatic, normal activity

WHO Clinical Stage III:

- Weight loss >10% of body weight
- Unexplained chronic diarrhoea, >1 month
- Unexplained prolonged fever (intermittent or constant), >1 month
- Oral candidiasis (thrush)
- Oral hairy leucoplakia
- Pulmonary tuberculosis
- Severe bacterial infections (i.e. pneumonia, pyomyositis)

And/or performance scale 3: bedridden <50% of the day during last month

WHO Clinical Stage IV:

- HIV wasting syndrome [i]
- Pneumocystic carinii pneumonia
- Toxoplasmosis of the brain
- Cryptosporidiosis with diarrhoea >1 month
- Cryptococcosis, extrapulmonary
- Cytomegalovirus disease of an organ other than liver, spleen or lymph node (e.g. retinitis)
- Herpes simplex virus infection, mucocutaneous (>1 month) or visceral
- Progressive multifocal leucoencephalopathy
- Any disseminated endemic mycosis
- Candidiasis of esophagus, trachea, bronchi
- Atypical mycobacteriosis, disseminated or pulmonory
- Non-typhoid Salmonella septicemia
- Extrapulmonary tuberculosis
- Kaposi's sarcoma
- HIV encephalopathy [ii]

And/or performance scale 4: bedridden >50% of the day during last month

The WHO accepts that their HIV staging system is several years old, and has consequent limitations in respect of which adaptations may be necessary. However, they still consider it to be a useful tool in deciding when therapy should be started in resource limited settings. The WHO recommends that all people who have WHO stage IV disease should start treatment. Making a decision about whether other people should start depends on what laboratory tests are available and in particular whether the person's CD4 cell count is known. A CD4 test measures the number of CD4 or T-helper cells in a person's blood.

The more CD4 cells there are per millilitre, the stronger is the immune system. The stronger the immune system, the better the body can fight illnesses.

So in summary, a person who has WHO Stage IV disease should start whatever the result of their CD4 test. They should also start if they have stage I or stage II disease and a CD4 count of less than 200. If the person has stage III disease, then whether they should start depends on their clinical symptoms, and it should also be taken into account whether they have a CD4 cell count of less than or equal to 350.

The Actuarial Society of South Africa (ASSA) defined the WHO stage model for South Africa based on the model presented by the WHO (Dorrington et al., 2006:4) and is summarized in table 2-2 below.

Table 2-2: Stages of HIV/AIDS by ASSA Model (Dorrington et al., 2006:4)

Stage	Description
1	WHO stage 1: Acute HIV infection
2	WHO stage 2: Early disease
3	WHO stage 3: Late disease
4	WHO stage 4: AIDS
5	Receiving antiretroviral treatment
6	Discontinued antiretroviral treatment

2.3.1. ART Treatment Programme worldwide

According to Nemes, Carvalho and Souza (2004), 120 000 Brazilians received antiretroviral therapy from 540 service sites throughout the country in 2004. Various studies were done in Brazil to examine the impact ART had on their HIV and AIDS pandemic (Saraceni, Da Cruz, Lauria and Durovni, 2005). Brazil introduced combined antiretroviral therapy in 1996 and all their drugs were given for free to the people. The number of deaths due to AIDS in Brazil fell 30.6% between 1995 and 1999. In total there was a 47.5% reduction in the number of AIDS deaths in Rio de Janeiro city from 1995 to 2003 alone.

Antiretroviral therapy in Haiti has been primarily provided through private initiatives. By the end of March 2005, 3919 people were receiving antiretroviral therapy in Haiti (World Health Organization, 2005).

2.3.2. ART Treatment Programme in Sub-Saharan Africa

By the end of 2004, 67 ART treatment programs were implemented in 20 different countries in Sub-Saharan Africa and 41,328 patients per year were treated by antiretroviral drugs between 2001 and 2004. Of the 67 programs, 16 programs (23.9%) were implemented by governments, 20 by NGOs (29.9%), 15 by private institutions (22.4%), 4 by academic institutions and foundations, (6.0%), 3 by international agencies (4.5%) and 9 by other institutions (Grigioni, Saba, Dintruff, Pechevis, Delbos, Muyingo and Ladner, 2004). In 2006 the antiretroviral treatment coverage was estimated at 23% among those with advanced infection (UNAIDS, 2006).

2.3.3. ART Treatment Programme in South Africa

Highly Active Antiretroviral Treatment (HAART) has been provided in South Africa for a number of years on a very limited scale (Stewart and Loveday, 2005). Provision was largely to the medically insured population through the private health sector and to some individuals receiving treatment through non-profit organization initiatives. However, the numbers treated were very small and did not address the need of the entire nation. In November 2003, after considerable sustained pressure from advocacy groups such as the Treatment Action Campaign, the government adopted the Operational Plan for Comprehensive HIV and AIDS Treatment and Care, which included the provision of ART in the public health sector. The Operational Plan outlined a multi-sector response to the pandemic, and specifically recognized the critical role of ARVs in the treatment of people with HIV/AIDS (Stewart and Loveday, 2005). It is envisaged that by 2009, all South Africans, including permanent residents, who require care and treatment for HIV/AIDS would have equitable access to ARV treatment (Stewart and Loveday, 2005).

The roll-out of the ARV treatment programme is proving a slow process. This is partly because the Department of Health needs to address major capacity and infrastructure constraints but also because it continues to broadcast confusing messages about the role of nutrition and **traditional medicine**.

Very little information was available on the number of patients receiving ARVs. The only reliable source available was the ASSA2003 projection model which is summarized in table 2-3 (ASSA2003, 2005).

Table 2-3: ASSA 2003 Projections of patients receiving Antiretroviral therapy

	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015
Eastern Cape													
On ART	1754	2979	10986	21371	34184	49030	65683	81043	95048	107709	119110	129387	138697
Discontinued ART	144	257	914	1866	3123	4652	6427	8191	9888	11487	12977	14360	15644
Free State													
On ART	2411	4002	8177	14379	22459	31822	42155	51084	58637	64894	69981	74050	77262
Discontinued ART	205	359	732	1318	2121	3100	4223	5290	6261	7114	7847	8465	8981
Gauteng													
On ART	8909	15369	40668	76514	122340	175489	234096	285214	328490	363935	391895	412991	428001
Discontinued ART	762	1383	3566	6951	11554	17178	23636	29826	35461	40379	44511	47858	50464
Kwa-Zulu Natal													
On ART	6634	10919	31296	58089	90587	126885	165816	199101	226981	249883	268415	283288	295209
Discontinued ART	562	976	2708	5234	8513	12377	16691	20740	24386	27569	30292	32594	34533
Limpopo													
On ART	3132	5154	9787	16371	24918	34865	46050	56327	65668	74092	81665	88488	94672
Discontinued ART	257	446	859	1480	2323	3352	4550	5738	6880	7955	8955	9883	10745
Mpumalanga													
On ART	3644	5891	10401	17511	26984	38033	50296	60860	69807	77270	83422	88462	92584
Discontinued ART	311	530	950	1629	2571	3727	5061	6326	7479	8498	9383	10141	10789
Northern Cape													
On ART	134	287	1542	2888	4312	5767	7220	8584	9835	10963	11961	12833	13586
Discontinued ART	11	25	126	254	402	562	728	892	1047	1192	1325	1444	1550
North West													
On ART	2162	3629	11210	20441	31171	42814	55053	65785	74978	82684	89018	94139	98225
Discontinued ART	183	324	962	1839	2935	4191	5564	6868	8057	9108	10016	10787	11434
Western Cape													
On ART	1389	5911	13498	22029	31393	41160	51085	60433	69013	76692	83398	89111	93849
Discontinued ART	116	487	1166	2015	3010	4100	5249	6383	7465	8470	9378	10181	10875
National													
On ART	26020	47434	123990	225775	351489	494731	651769	791001	912219	1015994	1103596	1176775	1237467
Discontinued ART	2194	4177	10757	20379	33022	48161	65381	81996	97405	111286	123539	134205	143407

2.3.4. ART Treatment Programme in the Free State

Following the South African National Policy (National Department of Health, 2003), the Free State Department of Health (FSDOH) established its Comprehensive Care, Management and Treatment of HIV and AIDS programme, which includes the provision of highly active antiretroviral therapy (HAART) on the 3rd May 2004. By the end of November 2004 the FSDOH had an operational facility, comprising a hospital and two clinics in each of its five health districts, a target set by the National Department of Health in the operational guidelines.

It was essential for the Free State Province that all levels of health care be involved from the start of the project in order to create a supportive environment which will include adults and children. With this, the hope would be that the ART service would be as close as possible to the community to ensure greater access and an increase in treatment compliance. The Free State Province therefore developed a framework model of care which was based on hospitals performing the role of treatment site while the local clinics performed the role of assessment site. Hence the entire region's resources will be used to help fight AIDS and support the ART treatment campaign.

Outlined below, was the proposed time frame for implementation of ARVs in the Free State (Chapman, 2003):

- Phase 1 (October 2003 - March 2004). Implementation at 3 ART Units (1 regional hospital and 3 clinics each, 2 district hospitals and 3 clinics each).
- Phase 2 (April 2004 – March 2005). Implementation at 5 ART Units (1 regional hospital and 3 clinics each, 4 district hospitals and 3 clinics each).
- Phase 3 (April 2005 – March 2008). Implementation at 16 additional ART Units (16 district hospitals and 3 clinics each) and expansion to 100 clinics.

By the end of September 2008, 26 009 adults have commenced on ART; of these 14 243 (54.8%) are still receiving treatment, 3 173 (12.1%) have died and 7 910 (30.4%) patients have been lost to follow-up. 683 patients statuses are unknown at this stage. By the end of September 2008, 2 326 children below 18 years have commenced on ART and 1 412 (60.7%) are still receiving treatment, 54 (2.3%) have died and 837 (36%) have been lost to follow-up. One should note that the high number of lost-to-follow up is due to a backlog of data that was not captured onto the system. The numbers were extracted from the data warehouse and its construction will be discussed in future chapters.

2.4. Free State Province Model of Care

The Free State model of care was structured around a nurse practitioner performing most of the treatment duties. In this model, clients or patients are screened (voluntary counselling and testing) at primary health care level (clinics) by nurse practitioners (called assessment service points). Those patients with a CD4 count below 200 or having AIDS already are referred to a nearby hospital for an initial assessment by a doctor (called treatment service points).

The doctor confirms whether the patient must go on ART. The patients are then referred back to the clinic for drug readiness training which usually lasts 3 weeks. On completion, they return to the hospital-based doctor for clinic review and initiation of treatment. Follow-up is primarily based at the clinic and nurses are responsible for 4 weekly distribution of medication dispensed by the hospital pharmacy, all blood monitoring (although results are reviewed by the doctor) and monitoring of adherence. Baseline assessment and 6 monthly follow-up assessments take place at the hospital by doctors who also issue prescriptions for ARV drugs. A standardized first line treatment regimen (Regimen 1a) of highly active antiretroviral therapy (HAART) comprising two nucleoside reverse transcriptase inhibitors (NRTIs; stavudine and lamivdine) and one non-nucleoside reverse transcriptase inhibitor (NNRTI; efavirenz) is provided to male patients and some female patients. For other female patients, an alternative NNRTI (nevirapine) is substituted for Efavirenz (Regimen 1b).

Table 2-4 summarizes the roles of the assessment points and treatment points. Table 2-5 shows a depicted model indicating the process that a patient will follow when he/she visits a clinic for ART.

Table 2-4: Free State ART Model of Care

ART Assessment Service Points (Primary Care Nurse Practitioners)	ART Treatment Service Points (Doctors)
• Voluntary Counselling and Testing • Clinical and CD4 staging • Serial CD4 monitoring for those with CD4 200 and no AIDS • Treatment of opportunistic infections. • Cotrimoxazole prophylaxis provision • Tuberculosis Screening • Drug Readiness Training • Routine ART Monitoring (distribution of drugs at 28 day intervals, draw monitoring bloods) • Adherence Support • Initial assessment of ART side-effects.	• ART Baseline Assessments (including exclusion of untreated tuberculosis, confirm eligibility for ART) • Investigation to exclude undiagnosed opportunistic infections (e.g. cryptococcal meningitis, extrapulmonary tuberculosis) • Management of severe opportunistic infections prior to ART initiation • Clinical review at planned ARV treatment initiation • Follow-up of patients on ART (review blood results, decide on single drug or regimen changes) • Management of severe ART side-effects

Table 2-5: Patients' Walk Through Model												
LEVELS OF CARE	1st contact	2 Weeks	3-4 Weeks	Next available (-3)	(-2 to -1 week)	(0 week)	(+1 week)	(+2 week)	(+4 weeks)	(+5 week)	Monthly	3 Monthly
Centre of Excellence	Expert consultation/ best-practice training inputs											
Regional Hospital	Specialty Referral											
District Hospital			Dr. review labs, history, order CXR, schedule for drug readiness							Dr. PE and lab review		Dr. PE and lab review
Clinic	VCT --> (+) Assessment (WHO stage 3/4, PE, CD4/ VL, Process information-basic pamphlets, support groups and link to community services) WHO stage 1/2 follow up 3 monthly re-assess	CD4 count < 200: Baseline labs, HIV/TB History, schedule Dr. assessment		Drug readiness training (1), sign treatment contract, meet with assigned community health care worker	Drug readiness training (2 to 3) start Bactrim	Drug readiness training (4) start ARV treatment	Nurse PE		Nurse PE and labs		Nurse PE and ARV refill	
Community					Home visit			Home visit				

2.5. Chapter Summary

This chapter provided an overview of the HIV and AIDS status worldwide as well as in Sub-Saharan African and South Africa. Antiretroviral therapy was introduced by the South African Government in an attempt to slow the spread of the diseases among the population. This courageous move formed the cornerstone of the Operational Plan for Comprehensive HIV and AIDS Treatment and Care in the South African public sector. The Free State model of ART care was examined which provided a good overview on the methodology followed by the Free State Department of Health on providing patients with the drugs.

The following chapter will examine the use of information technology to support the rollout of the ART programme. The existing decision system will be discussed with all its limitations and alternative ways-of-thinking that will be proposed to streamline the rollout of antiretroviral therapy.

CHAPTER 3 - INFORMATION REQUIREMENT OF THE ART PROGRAMME

3.1. Introduction

The previous chapter provided a detailed discussion of the worldwide impact of the HIV and AIDS disease. The usage of ARV drugs was examined as an intervention to ease the burden of the disease by prolonging the life of an infected patient. The "Model of Care" of implementing the ART programme was explained by referring to WHO standards and methodologies. The chapter concluded on how the "Model of Care" was adopted and implemented in the FSDOH.

In this chapter the information requirement of the ART programme will be discussed. This will be followed by a discussion of how the existing Hospital Information System was expanded to include a Clinic Information System in an attempt to answer the information requirements. Lessons learned during the implementation process and the lack of an integrated decision support system will be discussed at the end of this chapter. This chapter forms part of the *diagnosis phase* of the action research cycle.

3.2. Hospital Information Systems

According to the National Health Information System for South Africa (NHIS/SA) document, a Hospital Information System is conceived, from the outset, as a system that is fundamentally common to all hospitals and primary health care (PHC) centres and that it is an integral and a major part of the National Health Information System of South Africa (NHIS/SA, 2006). The document also states that any Hospital Information System (HIS) used in South Africa, should have the following core National Health Care / Management Information System (NHC/MIS) functions:

- patient registration
- a core, or a minimum data set, patient record
- appointment scheduling
- patient billing

Another key requirement is that the HIS should also be modular and that it should be completely developed. Simultaneously, a distributed Patient Database must be created and linked, via the Wide Area Network (WAN) to all the hospitals and Primary Health Care centres to both contribute to and use the database. The HIS should enable any of its features to be organised around a central database and must support different levels of access enabling restricted views of the centralised database.

The FSDOH honored these requirements when it implemented a HIS at three of its larger hospitals in 1996. The implementation was expanded between 2004 and 2006 with the inclusion of two more regional hospitals. All the hospitals using the HIS were feeding data into a central Patient Database, which in turn provided the ability to compile a minimum patient record. Taking this existing infrastructure into account, a decision was taken by the Free State Department of Health (FSDOH) Top Management team, that the Clinic Information System (CIS) to be used for all ARV clinics must be fully developed and linked with the existing HIS. The process to computerize the paper based information system will now be discussed and that will be followed by the CIS implementation.

3.3. Paper Based Information System

The backbone of the ARV data collection process in the Free State was built around the establishment of a formal paper-based information collection process first. This was done to assist the users with three fundamental elements:

- *Discipline to collect quality data*
- *Understanding of the ARV treatment programme*
- *Supplement the rollout of a Clinic Information System, once the users are familiar with the ARV treatment programme*

3.3.1. The Structured Records (the "forms")

Paper-based systems have long been the preferred medium for documenting clinical processes during consultations (University of Cape Town, Medical Research Council, Free State Department of Health, 2004(2)).

The FSDOH supported its continued use, given that it would be risky to impose an alternative system (like direct electronic capture) on doctors and nurses dealing with the clinical aspects of a **new** and **complex** treatment programme. It was proposed that the paper-based collection of data be combined with early capture into an electronic system, preferably at the source (decentralized mobile data capture) where the data capturer is able to validate responses with the clinician who completed the form. This "editing" process should be implemented right from the start of the programme to ensure that nurses and doctors understand how to complete the forms, and to encourage complete and accurate collection of data from the outset. Once unfavorable practices are established, it will be difficult to institute quality control measures to reverse them.

The following paragraphs will explain the contents and usage of each form and that is followed by a depicted illustration (figure 3-1) on the workflow of the structured records (the "forms") for collection of patient-specific data.

1. *Voluntary Counseling and Testing (Screen and Stage) Form*

   This form covers voluntary counseling and testing and CD4 staging. Patients may attend a clinic where VCT but not CD4 staging is offered. In this event the nurse could use the form as a referral to another clinic where CD4 staging is available. The form can also serve as a referral to the doctor at the local ARV treatment hospital. Referral criteria make provision for the referral of patients with CD4 counts ≥ 200 with an AIDS-defining illness or on current or previous treatment.

2. *Clinic Follow-up: CD4 ≥ 200 Form*

   This form is designed to record 6 monthly follow-up visits in patients with CD4 counts ≥ 200, but not yet on ARVs. It captures serial CD4 results. It has a strong focus on TB detection, and also makes provision for the documentation of cotrimoxazole prophylaxis in patients with late stage disease but CD4 ≥ 200.

3. *ARV: Hospital Baseline Assessment Form*

   This is the most detailed form and captures all the factors that may play a role in predicting treatment outcome. These factors include previous exposure to ARV, baseline CD4 and WHO stage.

4. *Drug Readiness Training Form*

   This form captures information on the three training sessions the patient attended at the assessment site.

5. *ARV: Clinic Follow-up Visit Form*

   This form is designed to be completed once a month when patients return to collect treatment. It documents weight, current medication and screens for drug-related side effects and intolerance. It also retains a strong focus on the early detection of tuberculosis.

6. *ARV: Hospital Follow-up Visit Form*

   This form is designed to be completed by a doctor at routine hospital follow-up visits. Initially these are planned at 3 month intervals but frequency is likely to decrease as patients become stabilized on treatment. The form makes provision for changes in opportunistic infection prophylaxis and ARV treatment.

7. *Referral: Clinic to Hospital Form*

   Standardized and tailored referral forms could play a vital role in ensuring good co-ordination of care between clinic and hospital. This form makes provision for 5 common reasons for referral to a secondary level, and prompts the primary care clinician to provide information needed for appropriate clinical decision-making at hospital level.

8. *Referral: Hospital to Clinic Form*

   This is a mirror version of the clinic to hospital referral form.

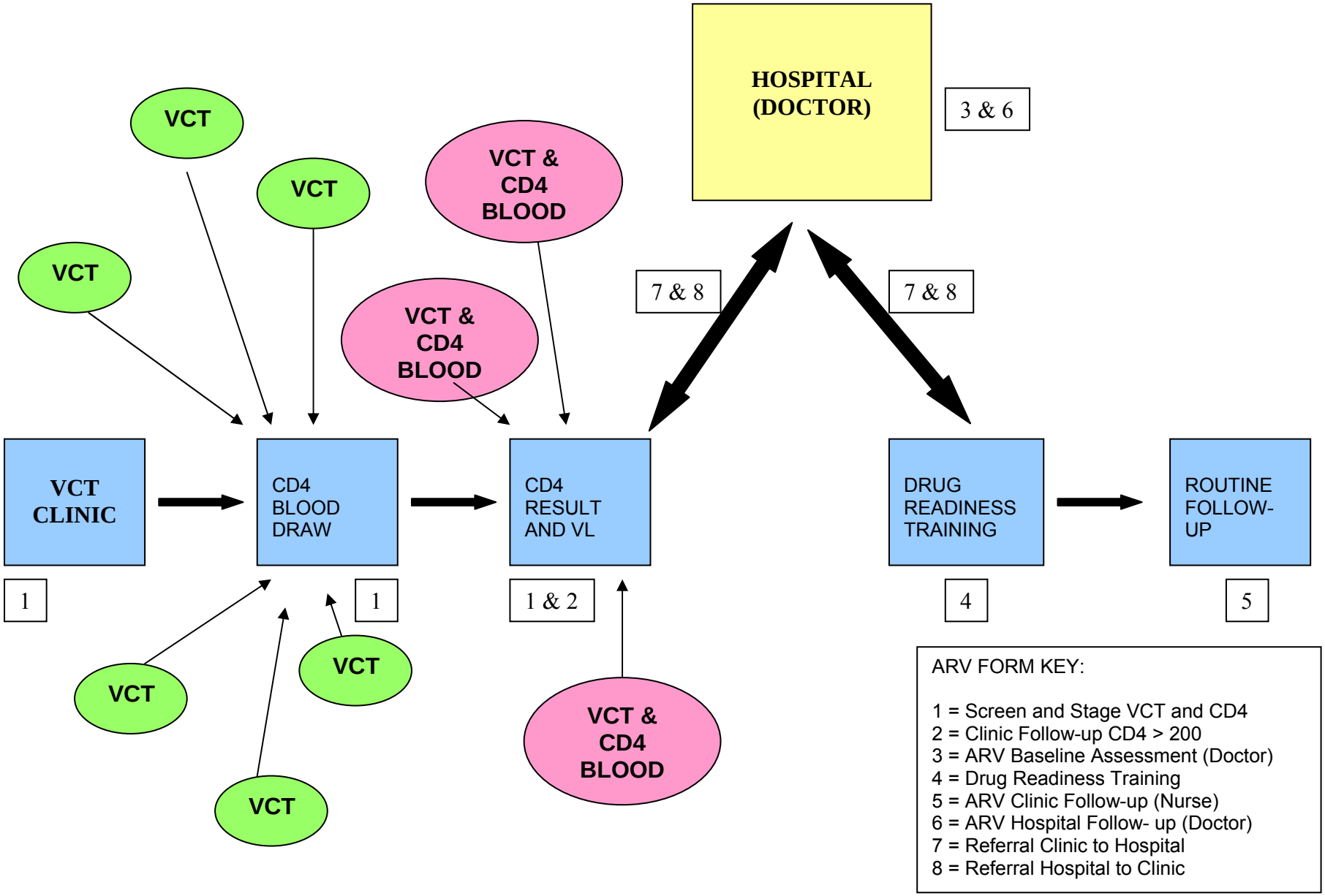


Figure 3-1: ARV Forms and Form Flow

3.3.2. Clinic Information System

In the previous section it was highlighted that one of the key requirement of a Hospital Information System is to provide all the hospitals and PHC centres the functionality to contribute and use the central patient database. Furthermore, according to the NHIS/SA document (NHIS/SA, 2006), for any Hospital Information System to be developed, installed and initiated in the major public hospitals, an appropriately scaled-down version of the same Hospital Information System must be implemented in all the smaller hospitals and PHC centres.

For this study, this system will hence be referred to as a ***Clinic Information System***. It was therefore imperative for the FSDOH that the existing Hospital Information System (Meditech) already used in its three major hospitals, be scaled down and implemented as the electronic system for all the ARV PHC centres. Lastly, the FSDOH had an additional requirement that all patient details fed into the central patient database, be accessible from any of the major hospitals already on the Hospital Information System. This would form part of the provincial wide initiative to create an online electronic health record of the patient. This exercise falls outside the scope of this study and will not be discussed.

3.3.3. Interim Palm© Pilot Handheld Computer Solution

Monitoring and evaluation is an important activity of the public-sector ARV treatment programme. For this purpose, structured records for collecting patient-specific information were developed, based on a paper information collection cycle. It was also proposed that this will teach the future users of the Clinic Information System discipline in terms of data collection and data validation.

As was highlighted in the previous section, in total eight paper forms have been designed and implemented at ARV assessment and treatment sites and hospitals throughout the Free State Province. All the forms are one page in length with the exception of the hospital baseline assessment form which is two pages long.

The MRC developed user defined screens on the Palm Pilot handheld computer which were based on the paper forms. The data capturers at each assessment and treatment site capture information that was filled in on the paper forms onto the Palm Pilot handheld computer. At regular intervals, data was exported from the Palm Pilot handheld computer and uploaded to the MRC via a modem and telephone line situated at each clinic. This was an interim data collection system and planned to be discontinued once the Meditech system was online and operational at all ARV assessment and treatment sites in the Free State. By the end of October 2006 all the Palm Pilots were phased out and replaced by the online Meditech system.

3.3.4. Meditech© Software

While the paper-based information system was in use and being captured by the interim Palm Pilot system, the acquisition and customization process for the Meditech software commenced. The initial investment of R2.5 million into the software licenses was justified by the fact that for sustainable long-term ARV monitoring and evaluation, the availability of electronic data available for analysis was pivotal. The customization process was not without its share of challenges and will be discussed later in this chapter. The first pilot version was only ready in November 2004, while the ART programme already commenced in May 2004. This meant that a backlog of paper would have to be captured while implementing the ART site at the same time.

Although the initial customization process delayed the implementation of all the Phase 1 ART sites, the Department of Health was able to recover lost time and had most of the sites using a Clinic Information System by the end of May 2005.

The software that was customized and implemented is called Meditech Medical Practice Management Suite (Meditech MPM) and was customized to accommodate all the paper-based forms and the ARV treatment process that was already in place. This software being one of the Meditech modules is actually a scaled-down version of the larger Meditech Hospital Information System and comes complete with the following functionality:

- *Community Wide Scheduling (CWS)*
- *Electronic Ambulatory Record (EAR)*
- *Authorization and Referral Management (ARM)*
- *Physician Desktop and Workload Management (PWM)*

3.3.5. Meditech© MPM Product Suite

The following section will briefly explain each of the sub-modules of the Meditech MPM product suite to give an overview of the system functionally and how this will be used in the future for clinical information gathering.

- ***Community Wide Scheduling (CWS):*** MPM manages appointment scheduling tasks by providing staff with booking options, check-in and registration tracking, links to patient profiles and conflict checking.
- ***Electronic Ambulatory Record (EAR):*** Electronic patient records enable nurses at the clinics to track important patient data, such as medications, lab results, referrals, progress notes and patient history. EAR offers a flexible data entry screen, that doesn't require a huge investment in training time, or a lengthening of the encounter process. EAR follows the already developed

paper-based workflow patterns that make sense for the physicians and nurses in either an ARV clinic or ARV hospital. The EAR consists of the following health care sections namely Chronic and acute medical problems, Medications (ARV drugs included), Allergies and Immunizations, CD4 count and Viral Load (Customer Defined Screen), Drug Readiness Training (Customer Defined Screen)

- ***Authorization and Referral Management (ARM):*** Integration of authorizations and referrals with registration, scheduling and recording of the patient's medical account are crucial. MPM offers integration with other MEDITECH applications and will contribute to the central Patient Master Index (PMI).
- ***Physician Desktop and Workload Management (PWM):*** MPM is equipped with a physicians/nurse desktop and workload management system that presents an interface to match a physicians/nurse workflow. Additional fields were added to accommodate the ARV paper forms, which include drug readiness training, viral loads, CD4 counts and lab results.

3.3.6. Meditech© MPM Software challenges

The customization and implementation process of the Meditech MPM software was not without a number of challenges.

One of the first challenges faced was during the customization phase. Meditech South Africa had to develop customer-defined screens (CDS) to accommodate the already developed paper-based forms. With this came the notion of building an effective dataflow mechanism that follows the paper-based system in place, but also adhere to best-practice software engineering principles such as user friendliness, screen-to-screen serialization and ensuring data integrity and the avoidance of data duplication. This challenge took almost six months and led to the initial delay in the implementation of the Meditech software. However, a win-win solution was proposed to accommodate the paper-based system in place, piggybacking on the user's familiarity with the complex ARV forms and data collection processes, while incorporating HIS functionality.

The second challenge was the issue of data quality. Because of the initial deployment of paper-based forms to both assessment and treatment sites in the province, numerous data elements of a patient were duplicated at each of the ARV PHC sites. With the implementation of the CIS, these duplicates were highlighted as the backlog was captured onto the system. For example, the same patient visited three different sites resulting in three different patient files and three different patient numbers. Subsequently, as each site captured the backlog, the same patient was captured three times into the master patient index, each institution unaware of the other's action. The only viable solution was to merge these records on a weekly basis and to systematically remove the inconsistencies within the system, created by the initial paper-based system. Users were also taught to make use of the master index lookup before creating a new electronic patient record.

Another concept introduced to improve data quality was to implement the Meditech software in the entire district at the same time. This included the treatment site and the referring assessment sites. With this method in place, all users were trained in the same week and also went live the same day. This also cultivated a culture of more efficient communication among the users in the same district.

The third challenge was implementing the Meditech MPM module itself. With the development of all the customer-defined-screens, the "out-of-the-box" functionality provided by MPM was overlooked. Instead the implementation methodology purely focused on using the customer-defined-screens. During the pilot implementation phase at the Motheo district in November 2004, it became very clear that this approach will not work as the MPM system functionality was now compromised. A strategy was devised to incorporate the "out-of-the-box" CWS functionality together with the EAR into all the customer-defined-screens. This new approach helped the assessment sites to schedule patient visits in advance and also to schedule their follow-up visit at the referral treatment site. This approach was adopted for all future site implementations and led to a reduction in capturing a patient's visit on the system.

The fourth challenge was to convince the FSDOH ARV Management team to include clinical staff (nurses and physicians) as users of the Meditech system. Most of the sites implemented only had the data capturer and the admission clerk as users of the Meditech system. While the original idea was to create a longitudinal ARV patient record by capturing the paper-based information, it had little or no clinical value now, since the clinical staff was excluded to validate data quality. After lengthy discussions among all the provincial role-players it was eventually decided that the implementation process will also provide PC's and printers for at least one nurse per site together with Meditech access and training.

3.4. ARV Data Warehouse

The MRC in collaboration with the University of Cape Town (UCT) Lung Institute developed a standalone ARV data warehouse for collecting information associated with the monitoring and evaluation of the Free State ARV and HIV and AIDS Treatment Programme starting in November 2004. The data warehouse schema was based on the data received from the Palm Pilots handheld computers that were used at the Free State ARV assessment and treatment sites. The data was uploaded via modem to a staging database situated at the MRC and then loaded into the ARV data warehouse. Although the data warehouse produced several managerial reports to the FSDOH ARV Management team, its use was discontinued due to poor data quality that was associated with the Palm Pilot system. The Palm Pilot system was discontinued and replaced with the Meditech Hospital Information system which was implemented at ARV assessment and treatment sites. The existing data warehouse schema was adopted to accommodate the new operational data source database structure by the MRC in January 2006.

To complicate matters, the FSDOH had a separate challenge in optimizing key internal business processes such as human resource management and revenue collection. Strategic information was required by the respective managers to improve the business performance for each of these business processes. Srivastava and Chen (1999) pointed out that the usage of data warehousing can result in reengineering of business processes to improve business performance.

The author of this thesis was tasked to develop a single data warehouse for the FSDOH. One of the objectives was to incorporate the standalone ARV data warehouse (developed by the MRC) as a data mart into the data warehouse. Other data marts such as human resources management and revenue collection were also required and had to be developed. The end product should be a single "appropriately designed" data warehouse made available for analysis to the relevant FSDOH managers. According to Scheese (1998) an appropriately designed data warehouse can be a cost-effective tool for business analysis and decision support.

3.5. Theoretical framework

A theoretical framework is proposed to solve the challenges that were presented in the previous section (3.4). The theoretical framework will conceptualize the two-phase action research methodology that was proposed in Chapter 1. Depicted below (see figure 3-2) is the outcome of the *diagnosis phase* for the first phase. It represents the Clinic Information System (Meditech MPM), the standalone ARV data warehouse situated at the MRC in Cape Town and the disparate interaction of groups of users with the two systems. The following section will cover the proposed theoretical framework for phases one and two.

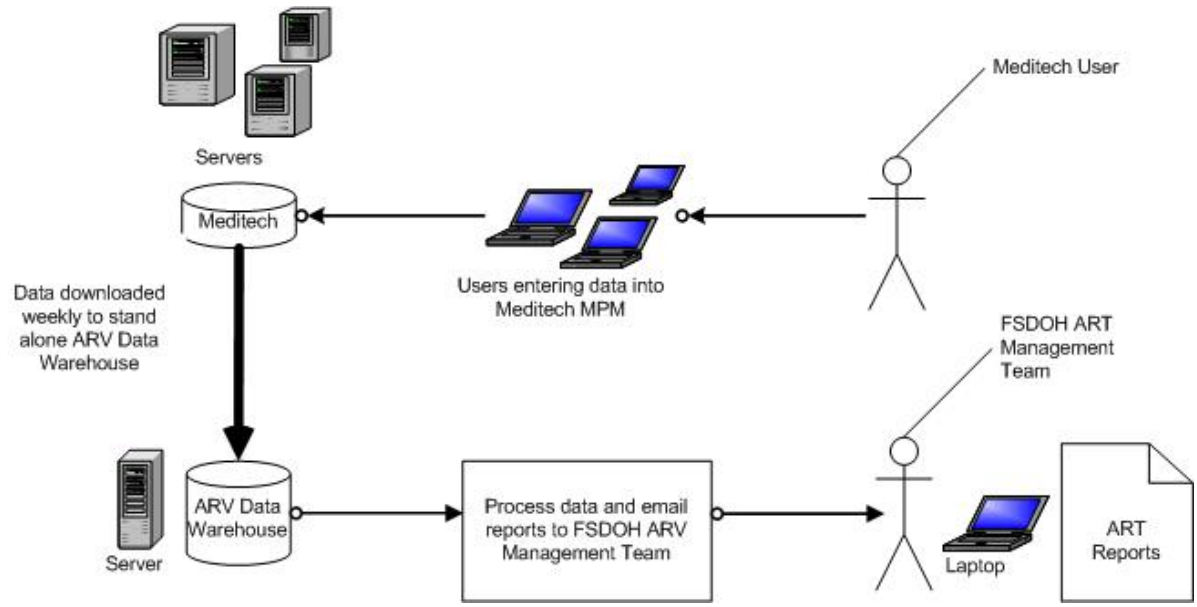


Figure 3-2: Outcomes of the Diagnose Phase for Phase One

3.5.1. Theoretical Framework for Phase One

The theoretical framework proposed for phase one will firstly comprise of converting the standalone MRC ARV data warehouse into a data mart and thereafter integrating it into the FSDOH data warehouse. Secondly, several other data marts will be developed and made available in the FSDOH data warehouse. These data marts will consist of crucial data elements that in turn will provide the necessary strategic information required to manage the ART programme. The construction of all these data marts will be covered in Chapter 6. In order to provide the necessary bus infrastructure common dimensions will be conformed as far as possible. This excludes the patient dimension. The reasoning for this approach will be covered in Chapter 4 as well as a theoretical overview of data warehousing. See figure 3-3 for the theoretical framework of phase one.

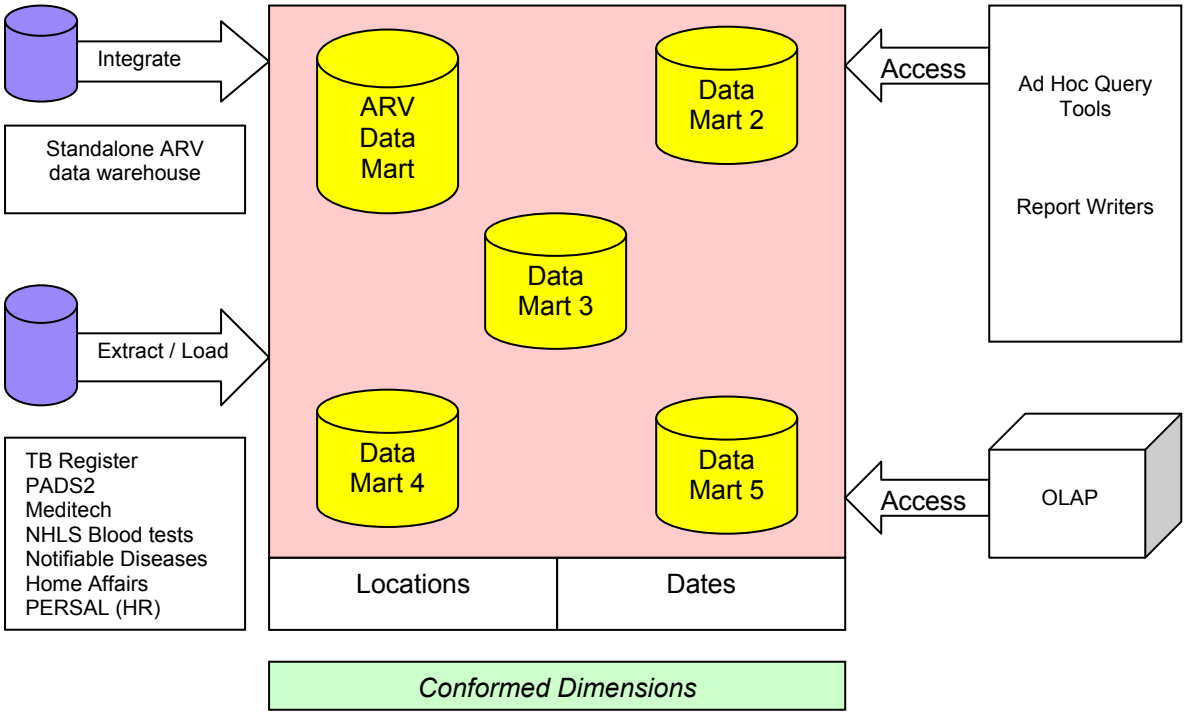


Figure 3-3: Theoretical framework for Phase One

3.5.2. Theoretical Framework for Phase Two

Phase two will comprise of supplying comprehensive integrated strategic ARV information from a longitudinal patient record. A theoretical framework is proposed to cover phase two. The theoretical framework will conceptualize the conforming process for all individual patient dimensions followed by the construction of the longitudinal patient record. Various probabilistic linkage mechanisms will be utilized to conform the patient dimensions (that was excluded in phase one). Chapter 10 will provide a theoretical overview of various probabilistic methodologies. Chapter 11 will provide the outcome of the probabilistic linkage mechanisms that was selected to assist with the conforming process.

Chapter 12 will provide insight into the construction of the logical as well as the physical longitudinal record. Various examples of the web interface that was developed to provide access to the integrated data warehouse will also be discussed. See figure 3-4 for the theoretical framework of phase two.

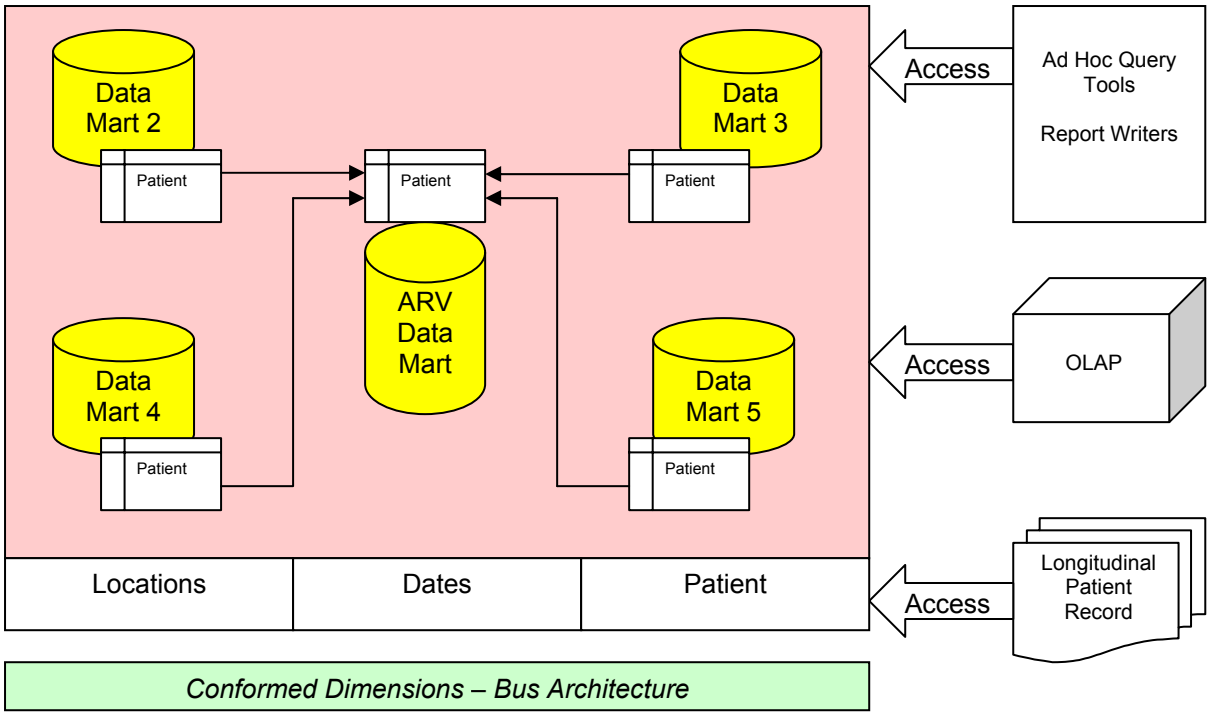


Figure 3-4: Theoretical framework for Phase Two

3.6. Chapter Summary

This chapter discussed the deployment of a patient-centric clinic information system which forms a vital part of providing information for strategic decision-making. An observation was made that in the preparation of a more sophisticated data capturing system, one needs to do the groundwork first and establish a good paper-based data gathering system. As the users grew accustomed with the system they also became more disciplined and started to provide higher quality data. This is an essential step to make any online clinic information system a success. Gradually the paper-based system was replaced over a period of four months with electronic devices and a central database. Input screens were developed that focused on the existing paper forms so that users could identify themselves with the previous data capturing system, turning electronic. A theoretical framework was proposed for the two-phase action research methodology, which in turn, conceptualized all the necessary steps and actions that will be followed to overcome the challenges presented in this chapter.

The next chapter will focus on the general theory behind data warehousing and how it relates to business intelligence. Using this theory the problem of converting the standalone MRC ARV data warehouse into a data mart and thereafter integrating it into the FSDOH data warehouse will be discussed.

CHAPTER 4 - MOTIVATION FOR THE SELECTION OF THE DATA WAREHOUSE ARCHITECTURE AND DESIGN METHODOLOGY

4.1. Introduction

The previous chapter covered the process of implementing the antiretroviral model of care using a paper based information system at first. The computerization of the paper based information system by a clinic information system was also discussed. The fact that it forms part of the larger hospital information system was highlighted. It was, however, noted that it still lacked basic strategic information. The MRC and UCT developed a standalone data warehouse to answer some of the basic strategic information requirements posed by the FSDOH. At the same time the FSDOH also identified the need for an integrated internal data warehouse. An *intervention* was required to prevent wasting valuable resources by having two data warehousing projects running in parallel to each other. A single **appropriately designed** and **integrated** data warehouse was required to provide a cost-effective business analysis tool for decision-making.

The purpose of this chapter is to provide the background knowledge for the forthcoming chapters on the usage of data warehousing design methods. The first section will define business intelligence and data warehousing. The data warehouse definition will be expanded with a discussion of the different data warehouse architectures and design methodologies. The chapter will conclude with a proposed solution for the FSDOH data warehouse architecture and how it differs from work already done in the healthcare field. This chapter fits into the *action planning* phase of the action research cycle

4.2. What is Business Intelligence?

Business intelligence (BI) was born within the industrial world in the early 1990's, to satisfy the manager's request for efficiently and effectively analyzing the enterprise data in order to better understand the situation of their business and improving the decision process (Golfarelli, Rizzi and Cella, 2004). The authors defined business intelligence (BI) as the process of turning *data* into *information* and then into *knowledge* and explains that knowledge is typically obtained about customer needs, the competition, general economic, technological and cultural trends.

Golfarelli et al. (2004) introduced an aspect of BI known as Business Performance Management (BPM) which can be defined as a set of processes that help organizations optimize business performance by encouraging process effectiveness as well as efficient use of financial, human, and material resources. Golfarelli et al. (2004) referred to Sonnen and Morris (2004) by pointing out that BPM includes a **data warehouse** as well as Online Analytical Processing (**OLAP**) tools, specialized Extraction-Transformation-Loading (**ETL**) tools and data integration systems. According to Fuchs

(2005), OLAP established itself as one of the pillars of a modern BI package and differs substantially from online transactional processing (OLTP) in terms of functionality.

4.3. What is a Data Warehouse?

The terms data warehousing and a data warehouse are often misinterpreted. Therefore, for the purpose of this study, the term data warehouse will be used as a noun and will refer to a system. Data warehousing will be referred to as the process of creating a data warehouse.

Inmon (2005:29) defines a data warehouse as a subject-oriented, integrated, nonvolatile and time-variant collection of data in support of management's decision. Inmon (2005:30) explains each of the parts of this definition:

1. Subject-oriented: Data is organized around major subject areas of the company. Example subject areas for an insurance corporation might be customer, policy, premium and claim. Each type of company has its own unique set of subjects.
2. Integrated: Data is fed from multiple, disparate sources into the data warehouse. As the data is fed, it is converted, reformatted, re-sequenced and summarized. The result is that data – once it resides in the data warehouse – has a single physical corporate image.
3. Nonvolatile: Data warehouse data is loaded (usually, but not always, en masse) and accessed, but it is not updated (in the general sense). Instead, when data in the data warehouse is loaded, it is loaded in a snapshot, static format. When changes occur, a new snapshot record is written. In doing so, a historical record of data is kept in the data warehouse.
4. Time-variant: Time variancy implies that every unit of data in the data warehouse is accurate as of some moment in time. In some cases, a record is time stamped. In other cases, a record has a date of transaction.

Kimball and Ross (2002:8,397) defines a data warehouse as the conglomeration of an organization's data warehouse staging and presentation areas, where operational data is specifically structured for query and analysis performance and ease-of-use.

The data warehouse is significantly different from a conventional operational or transactional database in several aspects. First of all, a complex data structure must be maintained in order to offer flexible and dynamic retrieval of rich decision-support knowledge (Shin, 2003). For this, it maintains a data architecture (Hristovski, Rogac and Markota, 2000) that is more integrated, subject-oriented, non-volatile and time-variant that transactional or operational databases (Dodge and Gorman, 2000:16; Hristovski et al., 2000; Shin, 2003). Data structures of a data warehouse should also be more cross-functional than "stovepipe" (Shin, 2003) and support management decisions (Hristovski et al., 2000).

It is important at this point to examine the relationship between a database, a data warehouse and a data mart. According to Hobbs, Hillson, Lawande and Smith (2005:4) a **data warehouse** is a **database** containing data from multiple operational systems that has been *consolidated,* integrated, aggregated, and structured so that it can be used to support the analysis and decision-making process of a business. Inmon (2005:370) defined a **data mart** as a data structure that is dedicated to serving the analytical needs of one group of people and states that the data mart structure will be fed from the granular data found (Inmon, 2005:132) in the **data warehouse**.

Beyer (2005) pointed out that the key differentiation between a data mart and the data warehouse is the intent of the logical model. Beyer (2005) explained that a data mart can also be subject-oriented, nonvolatile and time-variant but is usually de-normalized. Lastly a data mart can be *virtual* and does not have to be a physical database but can also be a logical set of views, cubes in a business intelligence tool, or a variety of other physical or logical constructs.

4.4. Data Warehousing Architecture

This discussion aims to give a holistic view of data warehousing and the high level data warehouse architecture. Both Shin (2003) and Orr (1996) defined the data warehouse architecture as a way to represent the overall structure of data, communication, processing and presentation that exists for end-user computing within the enterprise. Without the high level data warehouse architecture in place, the data warehouse will not exist. A literature study has revealed that two opposite points of view exist for data warehouse architectures.

Kimball Approach

Kimball and Ross (2002:7) give a graphic representation of the components making up a data warehouse. Figure 4-1 depicts the four distinct components of a data warehouse as the operational source systems, data staging area, data presentation area and data access tools. Data is extracted from the sources systems through a process called *data staging* or commonly referred to as *extract-transformation-load* (ETL) to the presentation area (Kimball and Ross, 2002:8). The data staging process involves four distinctive actions. Firstly, data is extracted from the source system and copied to the staging area. Secondly, the data is cleaned by correcting misspellings, resolving domain conflicts and dealing with missing elements. Thirdly the data is moved from the staging area to the data presentation area. Loading the data presentation area usually takes the form of presenting the dimensional tables to the bulk loading facilities of each data mart.

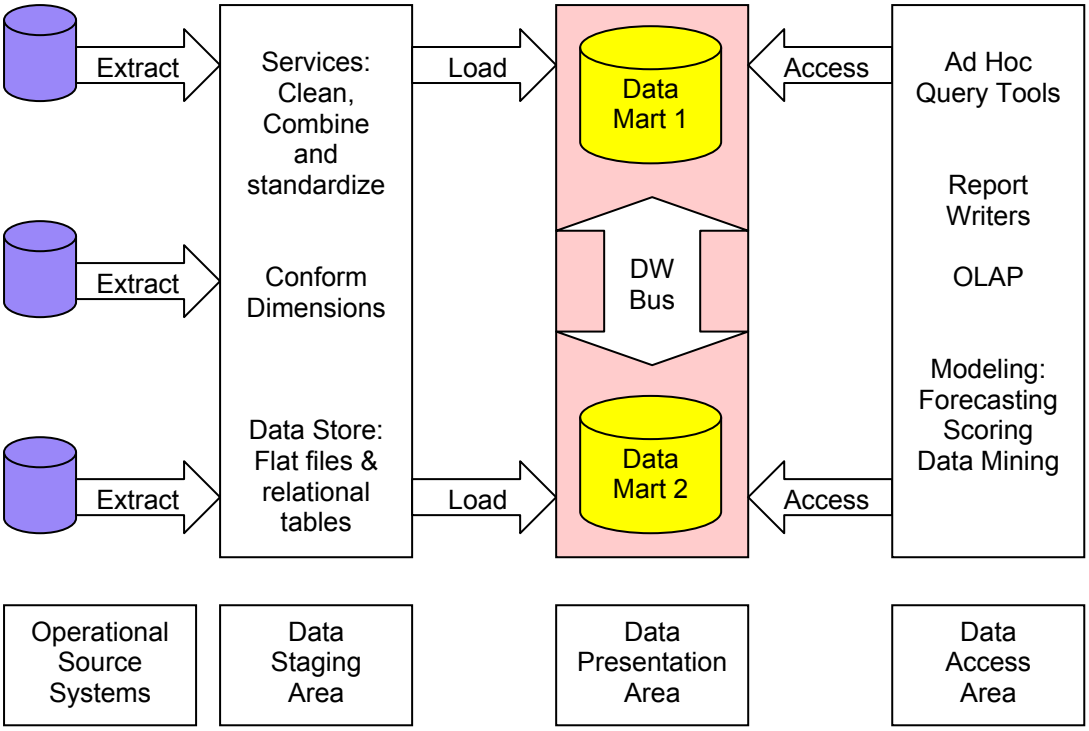


Figure 4-1: Kimball's view on a Data Warehouse Architecture (Kimball and Ross, 2002:7)

Kimball and Ross (2002:7) describe the data presentation area as a **collection of data marts**, each one based on a single business process but linked together on a data warehouse **bus architecture** using conformed facts and dimensions (see figure 4-2). Not adhering to a bus architecture (without shared, conformed dimensions and facts), a data mart is a "**standalone stovepipe**" application and will be the bane of the data warehouse project and would lead to a non-integrated data warehouse.

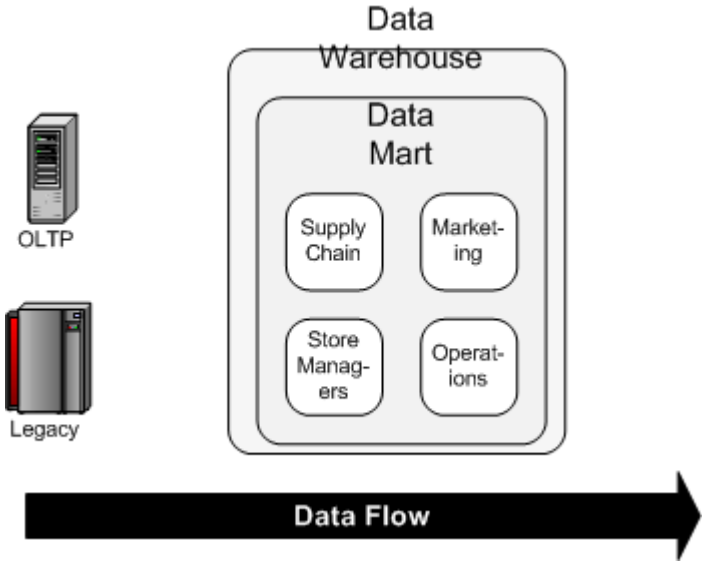


Figure 4-2: Kimball's Data Warehousing Design Methodology (Mailvaganam, 2007)

Kimball and Ross (2002:10) stresses the point that the data is presented, stored and accessed in dimensional schemas and should not be based on a normalized database schema. It is, however, acceptable to use a third-normal (3NF) relational format for the *staging area* but with the understanding that these normalized structures must be **off-limits** to user queries because they defeat understanding and performance (Kimball and Ross, 2002:10). The business case of using the bus architecture is advocated when building a distributed data warehouse system if there is a lack of resources (budget or time). When the bus architecture is used as a framework, the enterprise data warehouse can be developed in a decentralized (and far more realistic) manner (Kimball and Ross, 2002:13)

With the data presentation area (and bus architecture) in place, the data access area is the next important component. Kimball and Ross (2002:13) suggested that if the presentation layer is based on a **relational database** (for example Oracle), then these dimensionally modeled tables are referred to as star schemas. If the presentation area is based on **multidimensional databases** or online analytical processing (OLAP) technology then the data is stored in cubes. Kimball and Ross (2002:13) pointed out that most large data marts are still implemented on relational databases. In addition, most OLAP cubes are sourced from or drill into relational dimensional star schemas using a variation of aggregate navigation. All data access tools in the data access area query the data in the presentation area using either OLAP technology for cubes and ad-hoc query tools for dimensionally modeled tables.

Inmon Approach

Inmon's (2005) approach to data warehouse architecture differs from that of Kimball and Ross (2002). Kimball views the data warehouse as the sum of all data marts, each representing a single business process but using conformed dimensions and facts as depicted in figure 4-3. Inmon (2005:16) views a data mart as containing data derived exclusively from the data warehouse.

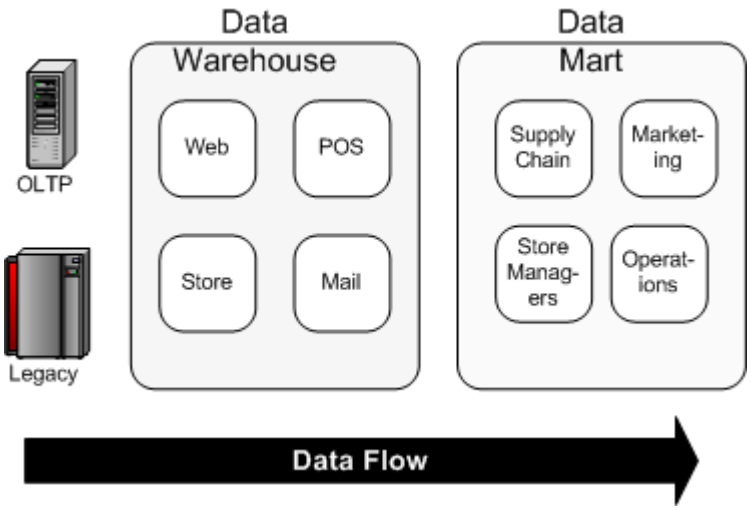


Figure 4-3: Inmon's Data Warehouse Design Methodology (Mailvaganam, 2007)

So in essence the main difference lies in th**e data presentation area** of the data warehouse architecture. From the literature study it was clear that the authors tend to agree on the other three components of the data warehouse architecture.

4.4.1. Data Mart design methodologies

Today there are several views on building data warehouses, with or without making use of the data mart design methodology. Srivastava and Chen (1999) argues that building **independent** ("end-run") data marts directly from legacy, external, and/or Operational Data Store (ODS) data files and databases **should be avoided**. It is best to first source the data into the data warehouse, thus becoming part of the "single source of truth", and then into a data mart, if necessary (Srivastava and Chen, 1999). Hobbs et al. (2005:11) opposes this argument and stated that it was discovered that many of the same benefits of a data warehouse could be scaled down to the department or line of business, solving a particular problem. The data warehouse would then contain multiple subsets that provide a consolidated enterprise view across all lines of business (Hobbs et al., 2005:11). This argument forms the basis of the evolvement and bringing into existence of data marts. The following section will discuss the different data mart architectures and design methodologies that exist today.

For the purpose of avoiding confusion it should be made clear that the different data mart approaches are more related to the Inmon style of data warehousing. Kimball avoids the issue by proposing the bus architecture of conformed dimensions and facts linking up all the data marts.

4.4.2. Independent data marts approach

Data marts can be dependent or independent, based on the source of information (Hobbs et al., 2005:12). The source of information for a *dependent* data mart is an existing data warehouse. A data mart is considered *independent* when no enterprise data warehouse exists (see figure 4-4a), and the data is extracted ***directly*** from the operational systems (Hobbs et al., 2005:12) and stored in data marts (Sweeney et al., 2002).

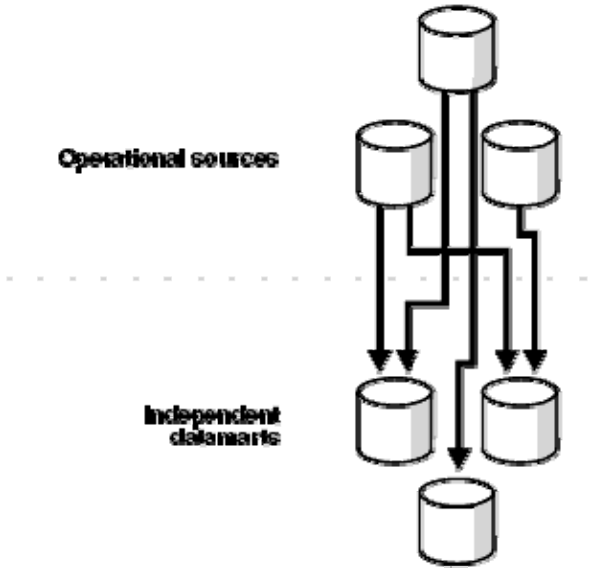


Figure 4-4a: Independent Data Mart Architecture (Lane and Lumpkin, 2000:20-4)

Because independent data marts can be constructed very quickly, they became quite popular in the mid to late 1990s. Only the data needed by individual departments needs to be identified and understood and a complete understanding of the corporate data is not necessary (Hobbs et al., 2005:12). Inmon (2005:370) mentioned that there is no need to "think globally" when building an independent data mart, but warn that independent data marts have the following problems:

- Do not provide platform for reusability
- Do not provide a basis for reconciliation of data
- Do not provide a basis for a single set of legacy interface programs
- Do require that every independent data mart create its own pool of detailed data, which is unfortunately causing redundancy.

The motivation behind independent data marts however can outweigh these above mentioned problems. The creation of independent data marts is often driven by the need to have a solution within a shorter time period (Lane and Lumpkin, 2000:20-6), which in turns builds the credibility of a data warehouse project. Inmon (2005:370) defended the usage of independent data marts by arguing that it is relatively inexpensive to build it and that it allows the organization to control its own information destiny.

4.4.3. Dependent data marts approach

The dependent dart mart architecture is the exact opposite of the independent data mart architecture. Dependent data marts (see figure 4-4b) receive data from an enterprise data warehouse ***before*** the data is shared with the business users. In other words, data is extracted from the operational systems, transformed and stored in an enterprise data warehouse. From there the information flows

to the data marts. In the previous section the problems of the independent data mart approach were highlighted and Inmon (2005:376) pointed out that dependent data marts will not have the same problems, as long as they take data from the enterprise data warehouse.

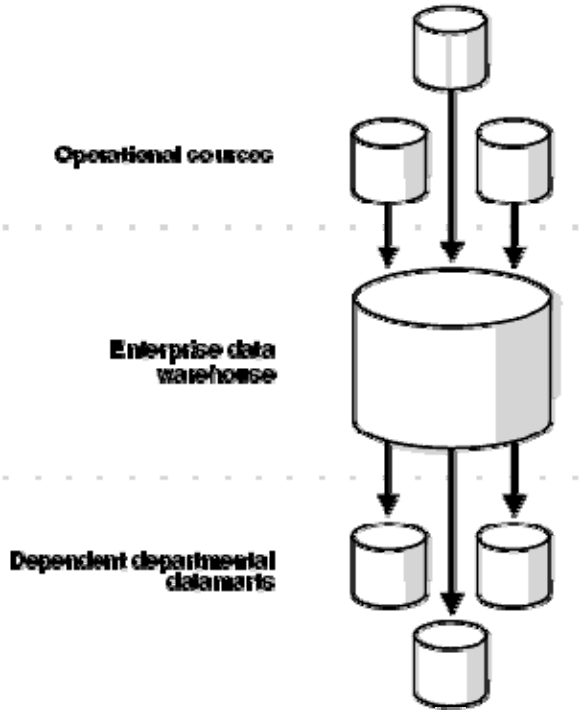


Figure 4-4b: Dependent Data Mart Architecture (Lane and Lumpkin, 2000:20-3)

According to Graig (1999) a dependent data mart will often contain only summary data. If a business user needs detail data, the software can generate the appropriate SQL to retrieve it from the data warehouse. For this architecture to function efficiently, one will need to build the data warehouse first which makes this approach relatively expensive.

The motivation for using the dependent data mart is to achieve improved performance and availability, better control, and lower telecommunication costs resulting from local access of data relevant to a specific department (Lane and Lumpkin, 2000:20-6).

4.4.4. Hybrid data marts approach

A hybrid data mart (see figure 4-4c) allows one to combine input from sources other than a data warehouse. This could be useful for many situations, especially when one needs ad hoc integration, such as after a new group or product is added to the organization (Lane and Lumpkin, 2000:20-5).

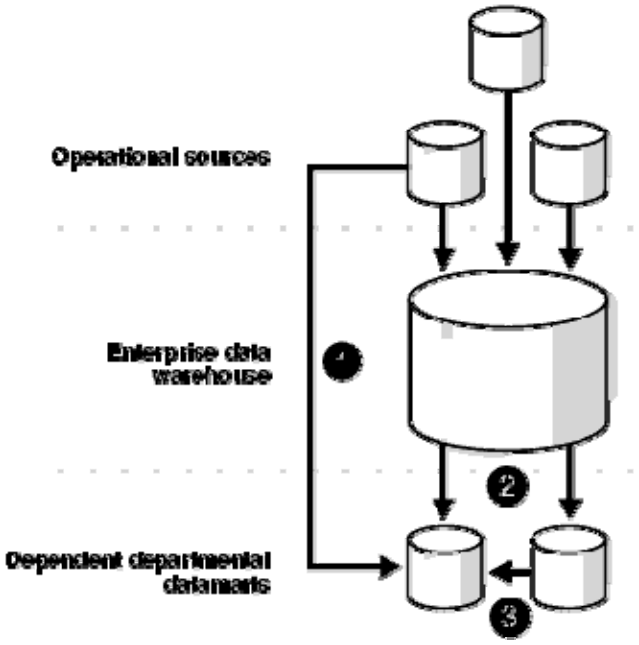


Figure 4-4c: Hybrid Data Marts Environment (Lane and Lumpkin, 2000:20-5)

In the above figure, process 1 illustrates the data marts receiving input from operational data sources, process 2 illustrates the data marts receiving input from the enterprise data warehouse and process 3 illustrates the data mart receiving input from another data mart.

4.4.5. Federated data marts approach

The federated data mart architecture combines the flexibility of an independent data mart with the discipline of dependent data mart architecture (Graig, 1999). Common ETL functionality is used to populate the data marts and the metadata is stored in a common repository to ensures uniform data access. Such consistency enables users to submit queries that bridge data marts, ensuring that the data will be meaningful and accurate (Graig, 1999).

4.5. Data warehouse data modeling

According to Shin (2003), several approaches exist today to design and implement a logical schema for data warehouse applications. These include star-schema design, de-normalization, fact and dimensional tables and different physical data structures such as indexes. More notable, only **two** possible data modeling styles are available to create a **quality** data warehouse. Both these styles were developed by the two main authors in the field of data warehousing namely William Inmon and Ralph Kimball.

4.5.1. Relational Model (Inmon style) approach

Inmon (2005:283) advocates the usage of a data-driven approach for gathering user requirements as design strategy for data warehouse modeling. List, Bruckner, Machaczek and Schiefer (2002) pointed out that the data-driven data warehouse development strategy is based on the analysis of the corporate data model and relevant transactions and it ignores the needs of data warehouse users a priori. The relational model approach is the most common data modeling style proposed by Inmon and uses database normalization to organize data into tables. This approach is the best long-term approach for building the data warehouse and for the case where a true enterprise approach is needed (Inmon, 2005:357). Inmon calls for a third-normal (3NF) relational format in which to store the extracted and transformed data (Lawyer and Chowdhury, 2004).

4.5.2. Multidimensional Model (Kimball style) approach

In contrast to Inmon, Kimball advocates the use of a requirement-driven approach for gathering user requirements as the design strategy for data warehouse modeling. Kimball proposes a four-step data warehouse design approach where he starts to choose a business process, takes the grain of the process, and chooses dimensions and facts (List et al., 2002). Kimball defines a business process as a major operational process in the organization that is supported by some kind of legacy system or systems (List et al., 2002).

Kimball's approach is also known as the **multidimensional model** or **star join approach** (Inmon, 2005:360), which views and models the data from a different perspective. Instead of considering an entity, which represents a thing such as a product or a place and the relationships between those entities, a dimensional model describes data using **dimensions** and **facts**, which becomes actual tables in the database (Hobbs et al., 2005:26).

A multidimensional model, as illustrated in figure 4-5 by an example, provides a very effective way of holding historical and current data in a form that makes it accessible to business users and that enables them to make the right business decisions. In this example, factual information was represented in the multidimensional model on customers placing orders for certain products from a telecommunications company. The dimensional model consisted of three dimensions (customer, product, time) and a single fact table (orders).

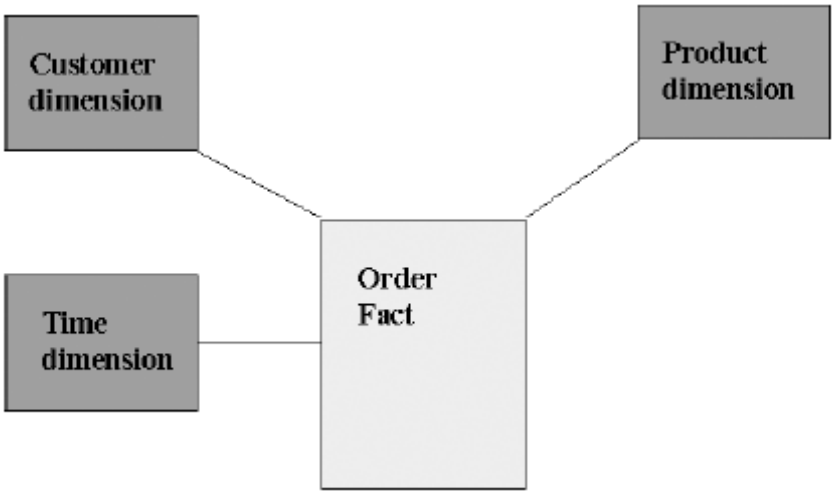


Figure 4-5: Example of a multidimensional model (Hobbs et al., 2005:26)

4.5.2.1. Dimension Tables

According to Berndt and Fisher (2001), dimensions define the query environment. Two of the important characteristics of dimensions are the richness of the attributes that describe the dimension and the hierarchical nature of the dimension. Ponniah (2001:212-213) states that a dimension table should contain a primary key to uniquely identify each row in the table. He furthermore explains that a dimension table is “wide”, meaning it will contain many columns or attributes and that most of the attributes will be of textual format.

4.5.2.2. Fact Tables

Hobbs et al. (2005:26) states that the fact table contains factual information and is usually the largest table in the data warehouse and is often growing the fastest. The information contained in the fact table doesn’t have to be at the finest level of detail; it could be summarized data. Generally the level at which data is held in the fact table is known as the *granularity* and is one of the important decisions the data warehouse designed must make during the design stage.

Berndt et al. (2003:371) quoted Kimball by pointing out that the best and most useful facts are numeric, continuously valued and additive. Both Berndt and Fisher (2001) and Ponniah (2001:215) stated that the most appropriate facts are *semi- or fully additive* numeric data items that can be summed, averaged or combined in other ways across the dimensions to form summary statistics.

The fact table will contain a primary key and according to Ponniah (2001:220) there are three options available when constructing a primary key for a fact table:

- *A single compound primary key whose length is the total length of the keys of the individual dimension tables*.

- *A concatenated primary key that is the concatenation for all the primary keys of the dimension tables*.

- *A generated primary key independent of the keys of the dimension table*. This key can also be called a surrogate key, even though the term is usually associated with dimension tables. Ponniah (2001:219) states that surrogate keys are simply system-generated sequence numbers and are used in **dimension tables**. According to Kimball (1998a), a surrogate is an "artificial or synthetic product that is used as a substitute for a natural product". Kimball (1998a) also argues it is a better design approach to use surrogate keys when joining dimension and fact tables instead of the actual physical keys (primary keys). Kimball (1998a) concludes that surrogate keys are just anonymous integers. These can be simple implemented by using synonyms (i.e. Oracle RDBMS) and then using a SQL statement to select the NEXTVAL from the synonym. If synonyms are used, one can also refer to surrogate keys as system-generated keys, but with no intelligence or hidden meaning associated with the key.

4.5.2.3. Factless Fact Tables

A "factless" fact table is a fact table containing only the concatenated primary key, with no measures or facts (Kimball and Ross, 2002:402) as table columns. Each fact table row will count as "one" for a fact. Kimball and Ross (2002:402) mentioned that a factless fact table is often used to represent events or provide coverage information that does not appear in other fact tables.

4.5.2.4. Fact Dimension Tables

Kimball and Ross (2002:269) pointed out that due to the extreme variability of medical record entries, a special dimension called *fact dimension* can be used. The authors depicted (see figure 4-6) that the entry type is a fact dimension that describes what the row means or, in other words, what the fact represents and argued that this approach is super flexible. New measures types can be added by just inserting a new row in the fact dimension table, instead of altering the structure of the fact table. Furthermore, NULL values are eliminated in the classical fact table design, because a row exits only if the measurement exists (Kimball and Ross, 2002:270).

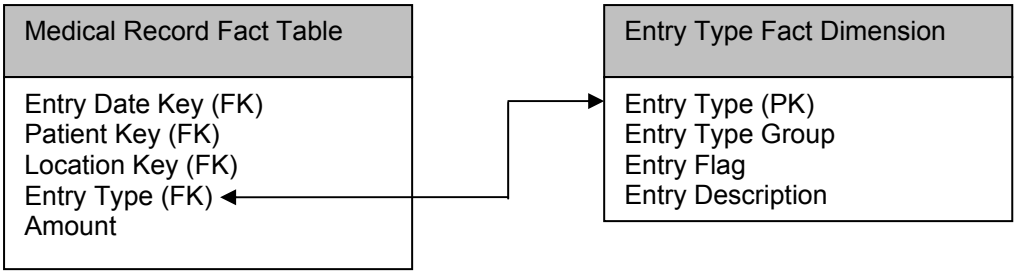


Figure 4-6: Fact Dimension

It is worth noting that Kimball and Ross (2002:270) warned that this approach can generate lots of new fact table rows, for example if an event resulted in 10 numeric measurements (fact measure columns), 10 rows will be in the fact table rather than a single row in the classic design. Furthermore, data access can be complicated when combining for example two numbers (that have been taken as part of a single event) which will now require two rows to be fetched from the fact table instead of one row using two column values.

4.5.2.5. Consolidated Fact Table

Kimball and Ross (2002:184) refer to fact tables that combine metrics at a common granularity as *consolidated* or *second-level fact* tables. Special attention should be paid when using consolidated fact tables by ensuring that it live at the same level of granularity and dimensionality just like the separate facts naturally live at a common grain (Kimball and Ross, 2002:185).

A common problem with using a consolidated fact table is **double-counting** and it occurs when two facts of different level of granularity is combined. For example one patient account with five detail account entries. If not carefully designed, the account amount will be repeated for each one of the five account detail facts, resulting in an inflated account amount instead of the correct account amount.

Knutsen (2005:23) suggested three practical methods to overcome this problem. During the construction of the FSDOH data warehouse a combination of these approaches will be applied.

Method 1: Add a column to the fact table and divide the fact amount by the number of detail facts

Order Num	Item Num	Shipping Charge	Total Price
1001	1	R5.00	R140.00
1001	2	R5.00	R60.00
		R10.00	**R200.00**

Example of Order = R200.00 with shipping charge of R10.00 for entire order using “Divide”

Method 2: Add a row to the fact table for each fact, using zeros in the unused fact columns

Order Num	Item Num	Shipping Charge	Total Price
1001	1	0	R140.00
1001	2	0	R60.00
1001	0	R10.00	0
		R10.00	**R200.00**

Example of Order = R200.00 with shipping charge of R10.00 for entire order using "Add Row"

Method 3: Add a row to the fact table using a fact dimension called Entry Type. Refer to 4.5.2.4 for an explanation of fact dimension tables.

Order Num	Item Num	Entry Type	Total Price
1001	1	ORDERCHARGE	R140.00
1001	2	ORDERCHARGE	R60.00
1001	0	SHPCHARGE	R10.00
			R210.00

Example of Order = R200.00 with shipping charge of R10.00 for entire order using "Fact Dimension"

4.5.2.6. Junk Dimension

Ponniah (2001:235) describes a junk dimension as a grouping of all other flags and texts that are meaningful together in a single "junk" dimension. These dimension attributes are useful for constraining queries based on flag or text values. Ross (2003) defined a junk dimension as a convenient grouping of flags or indicators to provide a recognizable, user-intuitive location for related codes, indicators and their descriptions in a dimensional framework.

4.5.2.7. Role-Playing Dimension

Kimball and Ross (2002:412) states that a role-playing dimension refers to a dimension can play different roles in a fact table depending on the context. Each of the dimension roles is represented as a separate logical table with unique column names through views.

4.6. Proposed FSDOH data warehouse architecture

Wood (2000) mentioned that there is a steady trend in the industry towards a **less risky**, **middle-of-the-road approach** in order to meet the requirement of healthcare professionals. Wood (2000) states that in this paradigm, the data model is designed using a top-down healthcare enterprise model, with the actual population of data implemented in a focused, iterative manner, typically starting with financial data, then moving toward encounter data, and finally to ancillary and clinical system data. As

scrubbed, audited data become available in the warehouse, independent data marts are spawned that serve as the distributed reporting base for the end users. Wood (2000) concluded that the advantage of this approach is that it offers the quickest path to return on investment (ROI) while providing flexibility to expand the data captured in the warehouse without redesigning the model.

The problem with this approach is the lack of resources required by the FSDOH to build a top-down enterprise healthcare model and a central data warehouse. Even with independent data marts, the effort required to construct the central data warehouse will be huge. One also needs to remember that the FSDOH management team was not familiar with data warehousing and did not see the anticipated business benefits initially. In order to move ahead, a political strategy was devised to target key business processes and to build independent data marts providing answers to them. This strategy would also allow for experimentation and prototyping to the researcher and at the same time would make the FSDOH management team aware of the business benefits using a data warehouse.

A combination of Inmon and Kimball methodologies will be used. Initially a data warehouse using independent data marts will be constructed to gain business buy-in. This is part of the prototyping phase. Human Resources and Revenue Collection business process will be targeted for this. Kimball's approach of star schemas will be used to construct each data mart. Due to the disparate and unknown nature of the operational data sources, it will be very difficult to construct a bus architecture at first. It is proposed to construct all these data marts and then to attempt linking (using conformed dimensions and facts) the bus architecture. The approach is illustrated in figure 4-7.

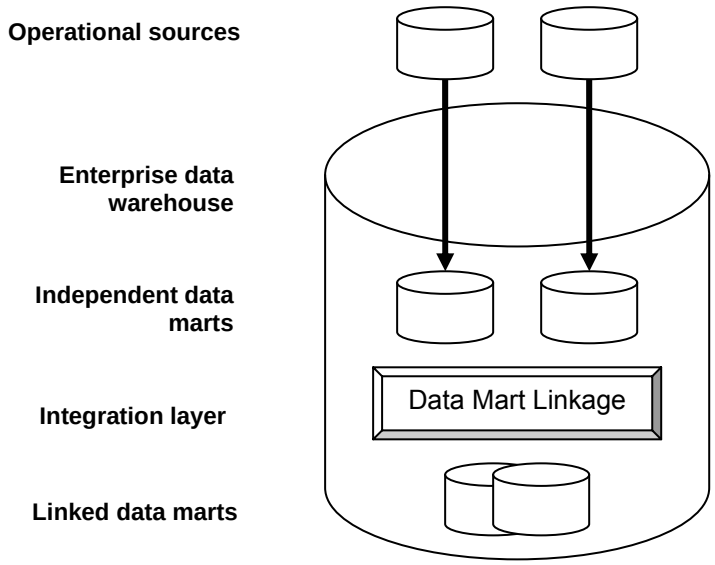


Figure 4-7: Proposed FSDOH Data Warehouse Architecture

Taking into account the benefits that the federated data mart approach offers, a business intelligence tool could be used to build a *federated BI solution* instead. At the end of the process the independent

data marts will be linked, conformed and provide a collection of data marts making up a virtual enterprise data warehouse as seen through the users of the BI solution.

4.7. Proposed FSDOH Hardware Architecture

The FSDOH opted to make use of the latest Redundant Array of Independent Disks (RAID) technology available and selected a Storage Area Network (SAN) to store the data for the data warehouse. Hobbs et al. (2005:91) suggested that using the *Stripe and Mirror Everything* (SAME) method for disks being mirrored and striped seems be the best storage configuration. Hobbs et al. (2005:92) also recommends a stripe width of 1MB and to place all the database files on these disks. The advantages of using a stripe width of 1MB can be summarized as follows:

- By striping all database files across all disks using a 1M stripe size, the use of the bandwidth across all of the disks is maximized and the occurrence of disk hotspots and bottlenecks are reduced.
- By mirroring, the availability of the database is increased by reducing the risk due to data loss from disk failure.

Taking all these recommendations into account, RAID 0+1 (striping and mirroring) will be used as the preferred storage configuration with a stripe width of 1MB on the EMC^2 CX-300 SAN. Millsap (1996) stated that RAID 0+1 provides excellent performance and excellent fault resilience, but with high acquisition cost. See figure 4-8 for the proposed technical layout of the SAN and the connection to the data warehouse server.

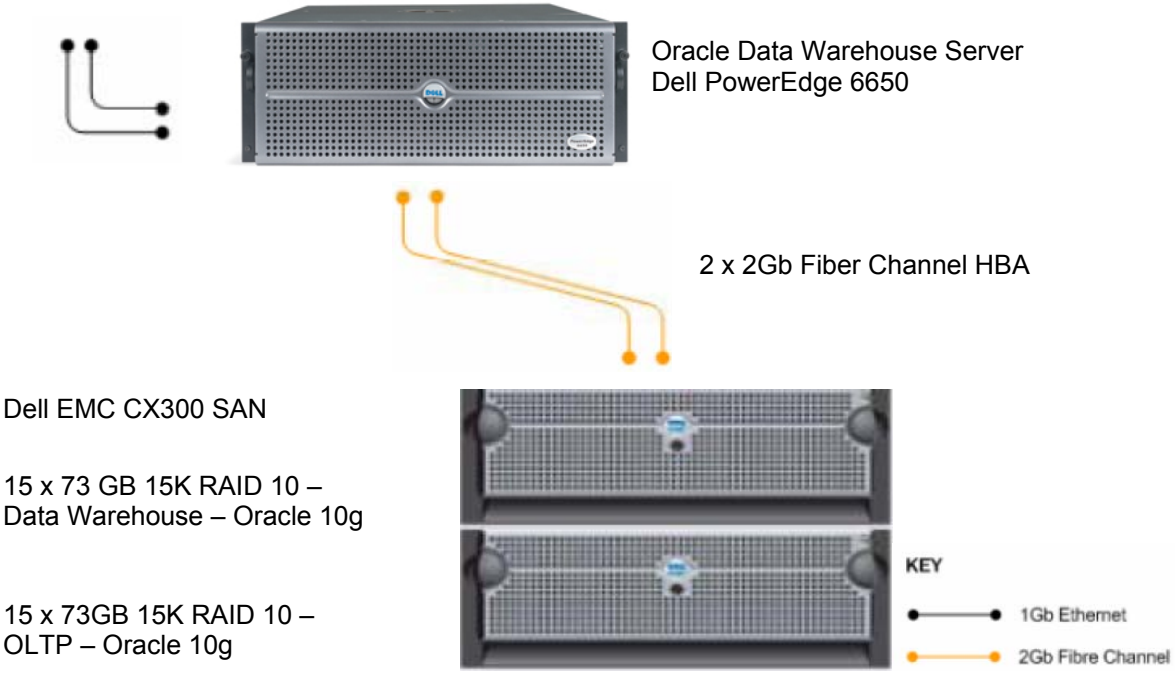


Figure 4-8: Proposed FSDOH Hardware Architecture

4.8. Proposed Skills Matrix

Kimball, Reeves, Ross and Thornthwaite (1998:739) proposed several technical roles for a data warehouse project team to develop and maintain a data warehouse. These roles can be summarized as follows:

- Project Manager (*)
- Business System Analyst (*)
- Data Modeler (*)
- Data Warehouse Database Administration (DBA) (*)
- Data Staging Designer (*)
- Data Warehouse Educator
- Technical Architect (*)
- Technical Support Specialist (*)
- Data Staging Programmers (*)
- Data Warehouse Quality Assurance Analysts

Due to the shortage of human resources and a very limited IT budget, the researcher fulfilled several of these project roles (indicated with an *). The researcher had to technically skill himself in Dell SAN architecture, Oracle 10g Data warehousing, Cognos 8 Business intelligence and GRLS probabilistic matching to build and maintain the data warehouse that was required. At the 2008 Gartner Symposium, Kelly (2008) quotes Hostman, an analyst from the Stamford, Conn.-based research firm who said that companies must build BI competencies from within their organization. Understanding internal data is what ultimately gives organizations a competitive advantage, and that's not something that should be outsourced. "You've got to own the competency, the program management, the skills development," Hostmann said. "That has to be a core internal competency."

4.9. Data Warehouses in Healthcare (General)

A literature study had revealed that despite the increasing role of the data warehouse as a strategic information source for decision makers (Shin, 2003), healthcare is lacking behind and in general has very little industry experience with ***enterprise-wide decision support*** (Ewen, Medsker and Dusterhoft, 1998) and the usage of data warehousing.

The following section will provide brief summaries of case studies found in the field of data warehousing and where it was specifically applied to the healthcare field:

- Lau and Catchpole (2001) proposed in their paper the usage of a data warehouse to improve surveillance and epidemiological studies in their field of monitoring sexual health in the United Kingdom (UK). They argued that the creation of the electronic patient record and, ultimately the

electronic health record is the key element to overcome the obstacles in clinical reporting. They proposed a data warehouse from disparate (clinical) databases and to use analytical techniques such as data mining to mine for new and potentially meaningful relationships and OLAP to allow users to do more complex queries.

  - *The work done by Lau and Catchpole (2001) is not relevant to this thesis because it focuses on monitoring sexual health in a developed country. Although they argued the creation of an electronic health record to be a key element in clinical reporting, it was not clear from their study whether the record was built and evaluated.*

- Hill Physicians Medical Group from California, USA constructed a clinical data warehouse and OLAP solution to assist with the processing and analyses of claims. The data warehouse included the following subject areas for the building of the data marts: Membership, Revenue, Primary Care Physicians and Claims (Verma and Harper, 2001).

  - *The work done by Verma and Harper (2001) is not relevant to this thesis because it focused on a different subject area (medical claims).*

- Hristovski et al. (2000) constructed a data warehouse and used OLAP technology to explore and analyze outpatient data at the Public Health Institute of the Republic of Slovenia. It was also envisage that the data warehouse would be expanded to include all inpatient (hospital visits) data as well. The authors concluded that building an outpatient data warehouse at the national level showed that data warehousing and OLAP are suitable technologies for building decision support systems in the domain of public health care.

  - *The work done by Hristovski et al. (2000) focused on general outpatient data while this study will examine outpatients, referred patients and inpatients involved in antiretroviral therapy.*

- Schubart and Einbinder (2000) conducted an evaluation study on the usage of a data warehouse at the University of Virginia Health System. The data warehouse was constructed to act as a clinical data repository (CDR) to provide direct access for clinical research and effective decision making. The authors concluded that users did not rate the current CDR highly in terms of Relative Advantage compared with other sources of information. Another finding was that the continued use of the CDR was strongly associated with an individual's work style and skills.

  - *The work done by Schubart and Einbinder (2000) used a CDR for clinical research while this study will provide a decision-making tool for managers involved in antiretroviral therapy.*

- Berndt (2001) used an existing data warehouse to explore the use of detailed data in constructing Consumer Decision Support Systems (DSS). OLAP technology was used to allow consumers to focus on particular characteristics or geographic areas rather than large-scale trends. The data warehouse supports standard community health reports and provides the detailed information required for in-depth investigations of important health care challenges. The data warehouse project has also evolved to support other capabilities, such as flexible community assessment at finer geographic levels and the creation of new health care indicators. One of the most important components of the data warehouse is based on *hospital discharge data.* According to Berndt, Hevner and Studnicki (2003) the Comprehensive Assessment for Tracking Community Health (CATCH) methodology played an important role in the development of the data warehouse and is often referred to in literature as the CATCH data warehouse.

    - *The work done by Berndt (2001) was done in a first-world country and did not deal with the same challenges as this thesis would encounter in a developing country. The study of Berndt (2001) also did not make any reference to the subject area of antiretroviral therapy and how the management of such a programme can be empowered with a DSS.*

- Nakache (2003) reports on a data warehouse application that was done for the French National Health Department. This project faced numerous problems including financial, medical, social, accounting, public health and political. The project's main challenge was, however, the issue of dealing with huge volumes of data. The work done by Nakache (2003) provided a methodology of solving the problem by using an ad hoc methodology mainly based on a data mart design.

    - *The work done by Nakache (2003) was done in a first-world country and did not deal with the same challenges as this thesis would encounter in a developing country. The study of Nakache (2003) also did not make any reference to the subject area of antiretroviral therapy and how the management of such a programme can be empowered with a DSS.*

4.10. Data Warehouses in Healthcare (Antiretroviral Specific)

- The Centers for Disease Control and Prevention (CDC) initiated several nationwide evaluation projects to develop a national evaluation system for monitoring and evaluating HIV prevention programs of health departments and community-based organization (Davis, Wan, Ross, Wen and Thomas, 2002). In order to meet this initiative a data warehouse was constructed to collect, manage, analyze and integrate large numbers of HIV prevention data. Using this data warehouse, the CDC and its HIV prevention partners could reduce the costs associated with assessing HIV prevention efforts and could perform more sophisticated analysis and modeling (Davis et al., 2002).

- *This study by the Davis et al. (2002) focused on the costs of HIV prevention efforts and is therefore not relevant to this thesis.*

- According to a research study performed by Chen, Accortt, Westfall, Mugavero, Raper, Cloud, Stone, Carter, Call, Pisu, Allison and Saag (2006), they were able to breakdown the costs of antiretroviral treatment using a data warehouse and statistical analysis using SAS Software version 6.1. Approximately 25% of patients at the Birmingham HIV Clinic had substance abuse and or mental health problems that warrant outpatient visits for therapy, resulting in the increase of outpatient costs that was not originally anticipated.

    - *This study by Chen et al. (2006) focused on the costs of antiretroviral therapy and is therefore not relevant to this thesis.*

- Snyman, Boucher, Cloutier, Puvimanasinghe and Ndwapi (2007) proposed the usage of a data warehouse to manage the implementation of antiretrovirals in Botswana by using only a subset of ARV treatment data (for example HIV prevention, costs of antiretroviral treatment, or simply a data warehouse with a de-normalized table) in a data warehouse.

    - *This thesis differs from the above mentioned study by proposing an integrated data warehouse which will be made up of ARV treatment data, human resources data, revenue collection data, tuberculosis data (closely related to HIV and AIDS), notifiable diseases data (closely related to HIV and AIDS) and patient hospitalization data (for patient follow up visits not linked to ARV*).

4.11. Chapter Summary

This chapter discussed the definition of data warehousing and how it relates to business intelligence. An overview was provided of the literature study done on the usage of data warehousing in healthcare and also in the specialized field of antiretroviral therapy. An argument was presented on why the FSDOH project differs from work done by other authors in the field. The data warehouse theoretical discussion included key design and architectural decisions and highlighted the reasons for opting to take certain avenues. Some of these key design decision included the proposed data mart architecture, data warehouse design strategy and the philosophical rationale of using dimensional modeling. The following chapter will look at an overview of critical Oracle 10g data warehousing features to be used for the proposed architecture.

CHAPTER 5 - DATA WAREHOUSE PERFORMANCE ISSUES

5.1. Introduction

In the previous chapter the theory of data warehousing was discussed together with the rationale of all the key data warehouse design decisions. A proposed FSDOH data warehouse architecture was introduced with a literature investigation on how this project will differ from work already done in healthcare and antiretroviral therapy management.

This chapter will examine key database features for maximizing the flexibility and performance of the FSDOH data warehouse. One needs to remember that *data warehouse designs* are quite different from the *highly normalized designs* that are typical in transaction processing environments. Many factors, such as the availability of data, the intended decision-making tasks, and the uncertainty of ad-hoc queries, influence the design process of a data warehouse. By investing in data warehouse performance features available in the database engine, some of these factors can be addressed. This chapter forms part of the *action planning* phase of the action research cycle.

5.2. Using Oracle to support the data warehouse infrastructure

In order to provide a high performance and scalable data warehouse infrastructure, Oracle Database Release 2 (Oracle10.2g) was selected as the data warehouse relational database management system (RDBMS). A brief theoretical discussion will be provided on the following database performance features (see figure 5-1), as pointed out by Stackowiak, Rayman and Greenwald (2007:98) to provide the necessary understanding.

Figure 5-1: Design Features for enhancing performance (Stackowiak et al., 2007:98).

5.2.1. Database Partitioning

Berndt et al. (2003:376) pointed out that partitioning is an important performance tuning technique and so is the use of the physical table partitioning. Baer (2005) confirms this fact by stating that the Oracle Database Release 2 ***with data partitioning*** can greatly enhance the manageability, performance and availability of almost any database application. Physical table partitioning involves dividing data into smaller, more manageable pieces allowing the optimizer to isolate a small portion of the total volume of a table for data access activities (Stackowiak et al., 2007:99). Partitioning a large table horizontally or vertically into small tables can improve the query performance by avoiding scans of a large table or by performing the table scan in parallel.

A fact table can be horizontally partitioned according to one or more dimensions say by product or by product and time (Wu and Buchmann, 1997). A fact table can also be vertically partitioned according to its dimensions i.e. all the foreign keys to the dimension tables are partitioned as separate tables. Obviously, in the distributed environment where the fact tables and the dimension tables are stored at distributed sites, a parallel semi-join can be easily applied by sending the vertical partition of the foreign keys to the dimension table (Wu and Buchmann, 1997).

Oracle provides several table partitioning methods to increase performance and manageability. The most frequently used partitioning types are: Range, Hash and List Partitioning (See figure 5-2).

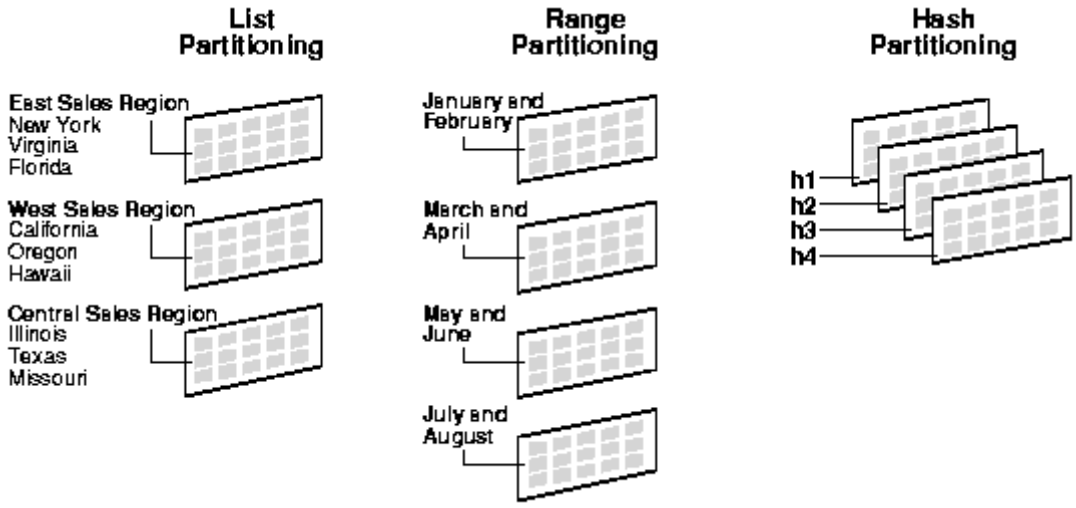


Figure 5-2: Table Partitioning Options (Cyran and Lane, 2003:18-5)

- **Range** — the most common partitioning method that allows the data architect to segregate data based on the ranges of data values or intervals within a column. Range partitioning is most commonly used for time series data (Stackowiak et al., 2007:99).

- **Hash** — Hash partitioning provides an alternative to range partitioning where data is evenly distributed across multiple partitions based on a hashing algorithm. Hash partitioning is most often applied generically to tables that lack time series data but possess significant volumes that require data to be equally isolated (Stackowiak et al., 2007:99).

- **List** — although list partitioning is a lesser-used partitioning method, it enables the data architect to address real-world challenges with the ability to divide data into discrete, finite subsets for organizational and management purposes. List partitioning is most often applied to data that can be segregated by organization, geography or other logical classifications (Stackowiak et al., 2007:99).

5.2.2. Database Indexing

Indexing is the most common approach used by data architects to satisfy performance requirements of a data warehouse. Indexing is leveraged to increase the overall efficiencies of data access by reducing disk I/O (Stackowiak et al., 2007:100). The three major indexing methods that Oracle provides to increase performance of a data warehouse are ***Bitmap Indexes, B-Tree Indexes*** *and* ***Bitmap Join Indexes***.

- **B-tree** — often utilized to enforce uniqueness for high cardinality data or provide enhanced performance for commonly searched or joined data (Stackowiak et al., 2007:100). Stackowiak et al. (2007:100) defined *Cardinality* as the number of unique data values in a column compared to the total number of rows and also state that B-tree indexes commonly provide the following:

    - Uniqueness of data enforced through primary key constraints or unique constraints.
    - Increased performance for foreign key relationships that have high cardinality to facilitate joining of tables.
    - Increased performance for one or more specific columns of high cardinality that are often searched.

- **Bitmap** — commonly used in a data warehouse or business intelligence solution where the column of data has low cardinality. These indexes are considerably smaller in size compared to B-tree indexes and provide faster response storing only the ROWID and a series of bits in a compressed format (Stackowiak et al., 2007:101). However, a modification to a bitmap index requires a great deal more work on behalf of the system than a modification to a B-Tree index and could influence a data warehouse negatively if not used correctly. According to Stackowiak et al. (2007:101), bitmap indexes commonly enhance performance for:

- Table joins where one or more columns have low cardinality (for example, in a dimensional model, joining a fact table that has many transactions to dimension tables used for look-up of parameters such as time and geography).
- Frequently searched columns that have low cardinality.
- Filtering conditions on specific columns that are NULL or contain equality conditions like AND, OR, and NOT.

- **Bitmap join** — used to pre-join foreign key values of a large table with primary keys of supporting tables providing a magnitude increase in performance (Stackowiak et al., 2007:101). Benefits include the following:

  - Pre-joining of common access paths that include filtering columns that can be resolved entirely within the index, thus eliminating the need for a physical table join.
  - Improved storage efficiency over traditional indexing methods.

5.2.3. Database Parallelism

One of the most effective methods for improving query access is to divide the work effort into multiple units of work that execute concurrently (Stackowiak et al., 2007:102). There are many parallel capabilities within Oracle, including parallel data loading, parallel Data Definition Language (DDL), parallel Data Manipulation Language (DML), and parallel query.

- **Parallel DDL** — provides efficiency when creating tables and indexes. One of the most efficient data preparation and dissemination operations is the CREATE TABLE...AS SELECT or CTAS statement (Stackowiak et al., 2007:103).

- **Parallel DML** — is useful for efficiently maintaining existing data. Can be used with INSERT, DELETE, UPDATE and MERGE operations, making parallel DML ideal for large batch operations to refresh or restate existing data (Stackowiak et al., 2007:104).

- **Parallel Query** — provides significant improvement in the speed of data retrieval operations. Parallel query can be initiated in a number of ways. The simplest method for invoking parallelization for queries is to enable parallelism at the table level. This can be done with an ALTER TABLE…PARELLEL 4 to create a parallel degree of 4 for example. The alternative method to invoke parallelism is to specify a *hint* as part of the query. A parallel *hint* specifies the table and the degree of parallelism to be initiated for a specified table provided there are sufficient resources (Stackowiak et al., 2007:104). The following statement serve as an example: SELECT /*+ PARALLEL(x,4) */ col1, col2 FROM test.

5.2.4. Database Summarization and Query Optimization

Oracle simplified the process of summarization by providing materialized views (mview), a schema object that provides a method to pre-join complex queries and performs commonly requested aggregates with full transparency to the end user or application developer. A materialized view physically stores the data that corresponds to the view's defined query (Dodge and Gorman, 2000:124). According to Goldstein and Larson (2001) query processing time can be improved through the use of materialized views.

In addition to improved query processing time, the materialized view structure is made transparent to business users and application developers by leveraging an optimization capability called query rewrite. A *query rewrite* operation occurs when the optimizer analyzes a query and determines if it can be satisfied by one or more materialized views at a lower optimization cost compared to accessing the base-level structures (Stackowiak et al., 2007:105). The optimizer will then rewrite the query (transparently to the user or application) to take advantage of the materialized view (Stackowiak et al., 2007:105).

Materialized views will also be used as ***static-joined dimension-fact tables*** in the FSDOH data warehouse. This approach reduces the join-time of expensive SQL joins during query execution time and will provide the business user with a simple structure. No *query rewrite* will take place, since the combined table forms part of the dimensional model.

5.3. Chapter Summary

This chapter provided the theoretical foundation of data warehouse performance features that will be used during the construction of the FSDOH data warehouse. By gaining an understanding into these features, the value of each will be understood when used as an intervention method to improve the performance of the FSDOH data warehouse. The following chapter will now examine and document the construction of the FSDOH data warehouse.

CHAPTER 6 - BUILDING A HEALTHCARE DATA WAREHOUSE TO PROVIDE STRATEGIC INFORMATION

6.1. Introduction

The previous chapter examined key database features for maximizing the flexibility and performance of the FSDOH data warehouse. This chapter will now examine and document the construction of the FSDOH data warehouse. This chapter forms part of the *action taking* phase of the action research cycle.

6.2. Business justification

With the implementation of the Integrated Health Planning Framework (IHPF) in the FSDOH during 2006, a gap in the coordination of all reporting needs in the FSDOH was identified by top management. Figure 6-1 provides an extract of the existing reporting structure and also highlights the lack of ***internal reporting*** to assist with tactical and strategic decision making. This was due to the fact that all the attention was focused on ***external*** statuary reporting requirements.

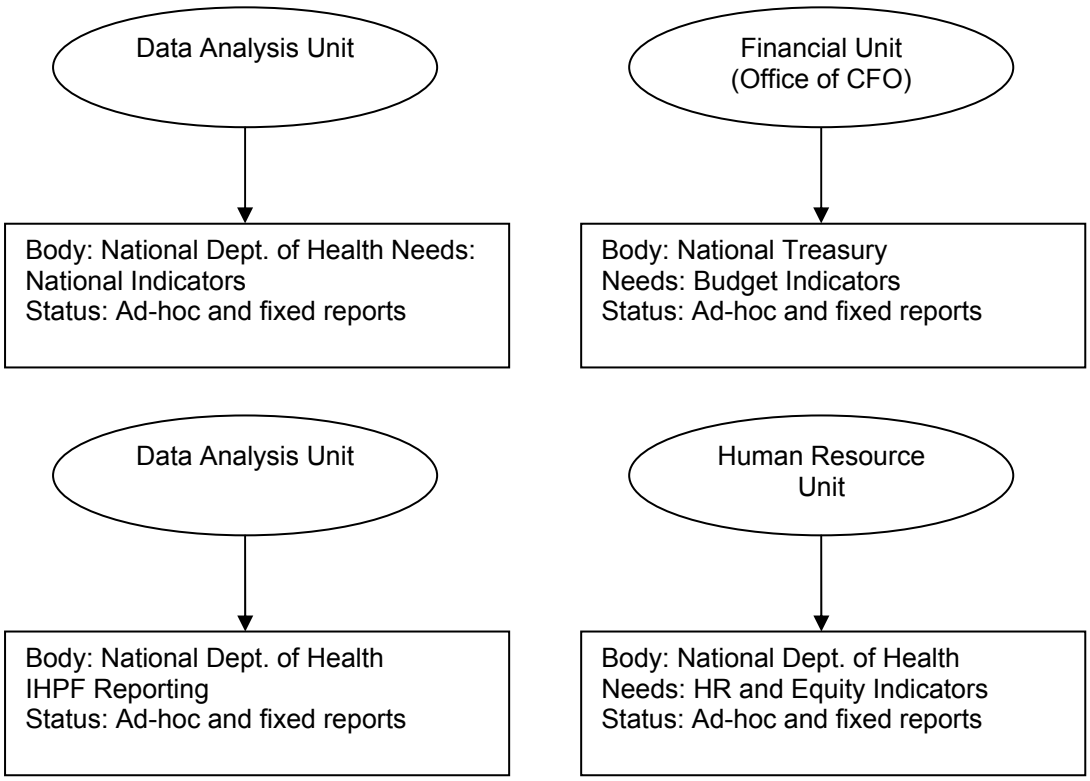


Figure 6-1: Current Reporting Framework

As an alternative, figure 6-2 proposed a different reporting structure that will satisfy statuary reporting needs, but at the same time provide top management and the Head of Department (HoD) with the necessary information to assist with strategic and tactical decision making. According to Srivastava and Chen (1999), all large organizations will need to analyze *long term data* for ***strategic*** data analysis and *day-to-day data* for ***tactical*** data analysis.

The proposed structure and information dissemination model highlighted the need for a ***corporate healthcare data warehouse***. All reports on indicators will be provided from it to ensure a single version of the truth, instead of every business unit trying to do their own strategic analysis. According to a research study done by Ewen et al. (1998), too much time is spent on gathering data instead of analyzing data. The FSDOH was no exception to this research finding.

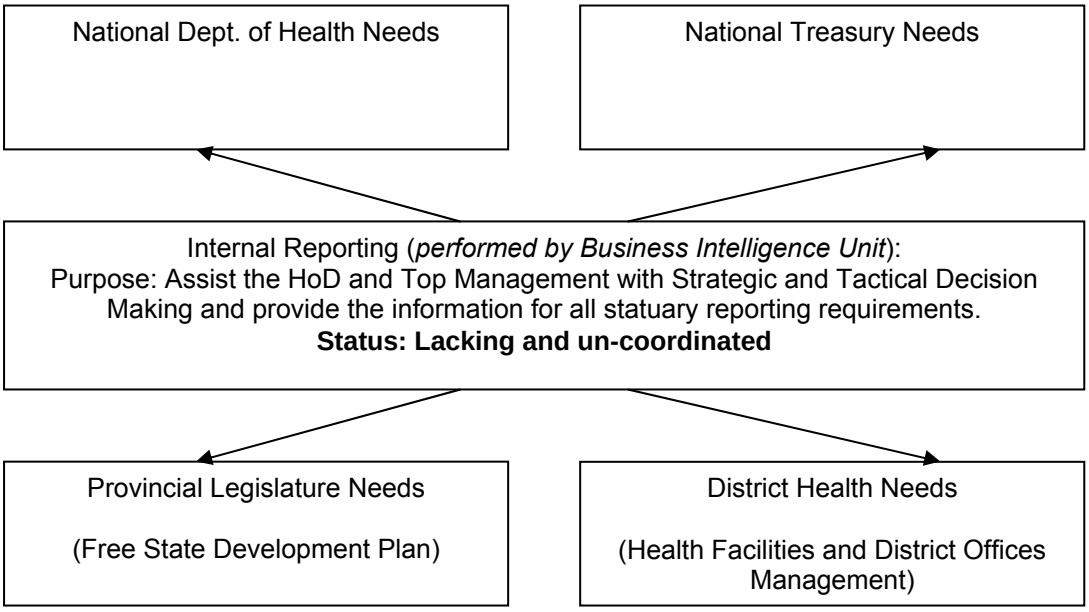


Figure 6-2: Proposed Reporting and Data Analysis Framework

6.3. Limited Resources to construct the Data Warehouse

From the previous section it is clear that a number of factors were forcing the FSDOH in the direction of a corporate data warehouse to provide a single version of the truth.

According to Schubart and Einbinder (2000), research has showed that the key factors for successful data warehouse implementation are organizational in nature. The authors also pointed out that management support and adequate resources are most important because these address political resistance. Gatziu and Vavouras (1999) stated that data warehouse development is a demanding and costly activity, with the typical warehouse costing in excess of $1m to construct. This can be a major obstacle in a developing country.

Taking these factors into account, top management was approached to direct the development of the FSDOH data warehouse. Because of a limited IT budget (0.68%), a decision was taken to break the project down into several data marts and to develop the data warehouse over a longer period of time. This would also reduce the risk of failure. Berndt et al. (2003:371) stated that the **star schema** or **dimensional model** has been recognized as an effective structure for organizing many data warehouse components. Chapter 4 discussed the rationale of using the proposed data mart architecture together with the dimensional model.

It was mentioned in Chapter 3 that the creation of a standalone ARV data warehouse was initiated by the MRC and UCT as a process to collect strategic information for the *monitoring and evaluation part of the ART programme*. This process happened in parallel with the creation of the FSDOH data warehouse. The remaining part of this chapter will discuss the development of several data marts and also the incorporation of the standalone MRC ARV data warehouse into the FSDOH data warehouse.

6.4. FSDOH Data Warehouse Overview

The following section provides a brief overview of the development of each independent data mart. It needs to be pointed out that the data marts were selected to addresses strategic priorities identified by top management. The development of these data marts forms part of Phase 1 of the FSDOH data warehouse project.

6.4.1. Human Resource Data Mart (HRDM)

The primary goal of this data mart was to provide the human resource development and management business unit with the necessary strategic information to perform workforce management for the entire FSDOH. The strategic information requirements will be listed in detail when the data mart development process is discussed. This data mart would provide the foundation for the ARVHRDM which will be discussed next.

6.4.2. Antiretroviral Human Resource Data Mart (ARVHRDM)

The primary goal of this data mart was to provide the managers in the antiretroviral programme with strategic workforce management information on staff that was involved in the ARV programme only. In essence this data mart is a subset of the HRDM, but focused on the staff employed at facilities involved in ART and not the entire FSDOH. This data mart would also provide strategic information required by National Treasury needed for funding HR costs associated with ART. The ARVHRDM addressed very specific needs while the HRDM addressed more generic needs.

6.4.3. Patient Admissions and Debiting Data Mart (PADSDM)

This data mart's primary goal was to gather strategic information relevant to revenue collection. With the intervention of integrating the ARV data warehouse, a secondary goal was identified to gather and link all clinical information of ARV patients that were admitted into hospitals for non ARV treatment.

6.4.4. Antiretroviral Clinical Data Mart (ARVDM)

The original ARV data warehouse provided ***clinical outcomes*** of the ARV programme as part of *monitoring and evaluation function*. With the conversion of the ARV data warehouse into a data mart and integrating it into the FSDOH data warehouse, secondary goals were identified to provide internal management reports as well as external reports for acquiring funding.

6.4.5. Tuberculosis Data Mart (TBDM)

This data mart's primary goal was to provide strategic information of the TB programme of the province. With the integration of the ARV data mart, a secondary goal was to link the two data marts together due to the close association of TB and HIV and AIDS.

6.4.6. Notifiable Diseases (NDDM)

This data mart's primary goal was to provide strategic information from the notifiable diseases transactional system in terms of tuberculosis and diarrhea which are both closely associated with HIV and AIDS.

6.4.7. Conformed Dimensions

By closely examining the data sources of the HRDM, ARVHRDM PADSDM, ARVDM, TBDM and NDDM data marts the following observations were made:

- All data sources required a standardized date table.
- All but two (HRDM, ARVHRDM) required a standardized location table.
- All data sources used different patient tables and it would require a major effort to create a standardized patient table. In order to get an operational data warehouse up and running as soon as possible, this task was postponed for a later stage.

To create a standardized table the conformed dimensional approach is followed. A conformed dimension is a dimension that means the same thing with every possible fact table to which it can be joined and generally this means that a conformed dimension is identical in each data mart (Kimball, 1998b). The conformed dimensions included the DATE and LOCATIONS dimensions. The only dimension that could not be conformed using traditional approaches was the PATIENT dimension. The following paragraphs will describe the conforming processes followed for the DATE and LOCATIONS dimensions. The process to conform the PATIENT dimension will be covered in Chapters 10 to 12.

6.4.7.1. Conforming the Dimensions

The process to create a conformed location dimension was a relatively straightforward one. The argument was made that the dimension table will be small and will contain approximately 2000 records. By closely examining each operational data source, it was discovered that different facility names and keys were assigned to a similar location. The reason for this is obvious – each operational data source was developed in an isolated manner and it was never envisaged that these operational data source will need to talk to each other in the future. To avoid different versions of LOCATION information, a single bridge tabled called TREATMENT_LOCATION_BRIDGE was developed and implemented in the data warehouse.

This bridge table will be used and shared between the PADSDM, ARVDM, TBDM and NDDM data marts. Furthermore the table will also introduce the formal conforming process of standardizing treatment facility names. Another advantage was the introduction of a universal treatment primary key that could be used in all the different data marts. See figure 6-3 for the layout of the TREATMENT_LOCATION_BRIDGE table.

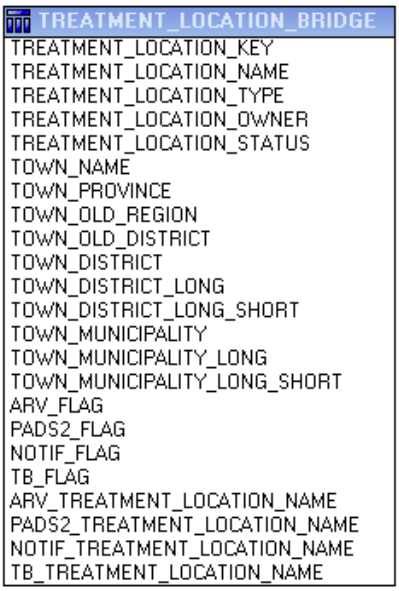


Figure 6-3: TREATMENT_LOCATION_BRIDGE "helper" table

The ARVDM and ARVHRDM will not be using the TREATMENT_LOCATION_BRIDGE table, because data is arranged using the traditional business units hierarchy and not by facilities. This difference will be covered in detail in the section that describes the construction of the HRDM. By using the implemented TREATMENT_LOCATION_BRIDGE, a conformed dimension table called TREATMENT_LOCATION_DIM was constructed in the data warehouse. This conformed dimension was used in all the applicable data marts. The DATE dimension was pre-generated and populated with all possible date combinations ranging from 1950 to 2050 and was used in all the data marts. See figure 6-4 below for the relationship of the conformed dimensions and the data marts.

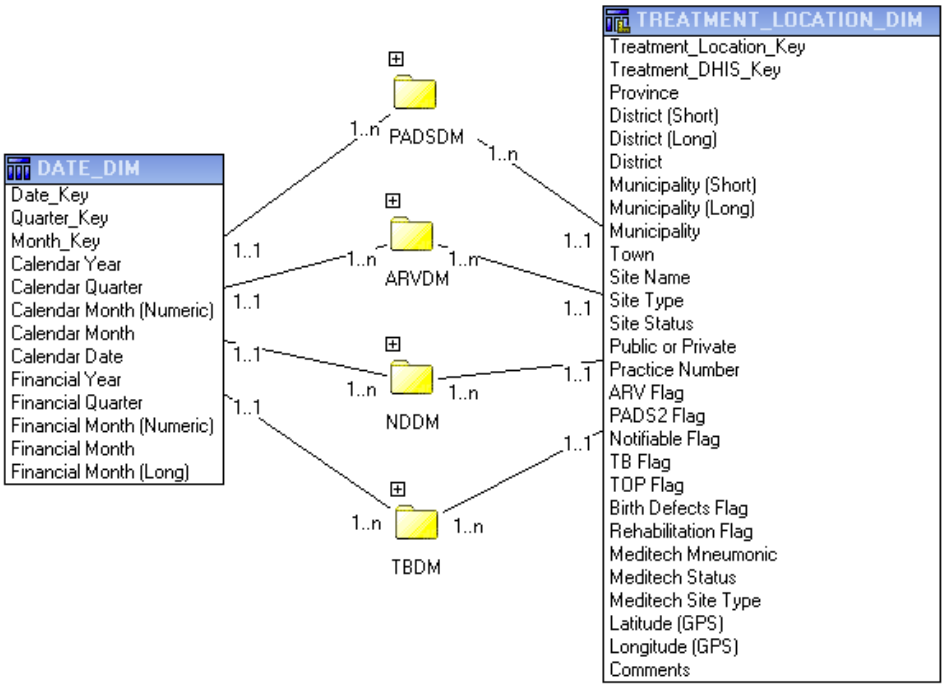


Figure 6-4: Conformed dimensions

6.5. Development of the Human Resources Data Mart (HRDM)

Most companies and organizations today are performing some type of human resource (HR) metrics based on employee counts, such as turnover ratio and average tenure. With HR metrics, usually the collected data is the ending point. Once the data is gathered, the workforce has been summed or counted. However, the main tool in providing human capital management is workforce analytics (Barrette, 2004).

If one takes these statements of Barette (2004) and apply it to the FSDOH, it means that workforce analytics is just as important to the FSDOH as most outside companies. The challenge however, is the lack of a workforce analytics system to measure HR metrics. The FSDOH like all other National and Provincial Government Departments only have access to an online transaction processing (OLTP) based payroll system. This payroll system is called the ***Personnel and Salary System*** or more commonly known as ***PERSAL*** in South Africa. The system has been in a production environment since 1990 and was developed in Natural Adabas. During the time of this study, the maintenance on the system was done by a private company.

Because the system is OLTP based, it proved inadequate in providing the necessary human resource statistics. The OLTP system solely focused on compliance reporting and day-to-day queries. It was not designed to perform workforce analytics. Furthermore, every new change or new report must be submitted to a central *System Change Control* system. From there it will be prioritised and once accepted, handed over to the private company for development. This process was cumbersome, inflexible and took forever to happen and led to overall frustration. In 2004 the FSDOH received approval from National Treasury to extract all relevant human resource data from PERSAL, allowing them the freedom of loading the data into a data warehouse.

6.5.1. Addressing Managerial Outcomes

Listed below are the outcomes of the HRDM that was identified as key strategic information requirements:

- Staff head count and vacancy rate
- Funding vacancy financial implications
- Absenteeism rate
- Qualifications distribution

6.5.2. Extraction, Transformation and Loading Challenges

6.5.2.1. Data Extraction Process

The data in the OLTP system (PERSAL) is *transient data* of nature. According to Bruckner and Tjoa (2002), the key characteristic of *transient data* is that alterations and deletions of existing records physically destroy the previous data content. In order to keep the history of the data in tact, all modifications to the data had to be considered.

The ETL processes commenced with a data extraction process which are performed **twice** a month by Treasury. At the beginning of every month, the FSDOH will receive two sets of data. The reason for this approach is entrenched in the manner government officials receives their salaries in South Africa. Permanent staff is paid the 15th of the month. All the information related to this event constitutes data set one. However, additional or supplementary payments (i.e. S&T claims, overtime, fuel allowance, etc.) and workforce operations (promotions, staff re-allocations) can be made to government officials between the 16th and the end of the month. All these additional information constitutes data set two.

In essence, the first extraction consisted of a full *data snapshot* taken on the 15th of the month from the *transient data set*. This included **staff, posts** and the **hierarchical organizational structure**. According to Bruckner and Tjoa (2002), a *data snapshot* is a stable view of data as it exists at some point in time. It is a special kind of periodic data. Snapshots usually represent the data at some time in the past, and a series of snapshots can provide a view of the history of an organization.

The second extraction consisted of the supplementary changes to the *data snapshot* picture of the first extraction set in terms of **staff** and **posts** but **excluded** any changes to the hierarchical organizational structure. This extraction process was preformed at the end of the month. It can be seen as *semi-periodic data*. According to Bruckner and Tjoa (2002), almost all operational systems keep only a small history of the data changes due to performance and/or storage constraints.

The challenge with more than one set of extraction data in the update window is the issue of *late-arriving data.* According to Bruckner and Tjoa (2002), *late-arriving data* are bothersome because they are difficult to integrate with existing fact and dimension tables, especially when surrogate keys are used in order to cope with slowly changing dimensions. The newly integrated datasets will also change the counts and totals of prior history. Late-arriving data can therefore possibly change analysis results unexpectedly from the analyst's perspective.

In order to deal with the problem of late-arriving data, it was agreed that the data warehouse will be updated in the **first week** of the following month, reflecting the *transient data picture* and supplementary changes (*semi-periodic data)* that was made to it.

6.5.2.2. Time Stamping

The standard approach for storing periodic data (typically found in Data warehouses) is to use time stamped status fields for each record. For the HRDM the *load timestamp* method was used.

Slow changing dimensions (SCD) Type 2 will be used as far as possible. According to Berndt and Fisher (2001), this type of change adds rows to maintain an arbitrarily long history. The keys must be "generalized" in this approach by using a version number or some other mechanism, so that related rows can be retrieved as a coherent history. Each table in the staging area had a column added called EXTRACT_DATE which translated to the record *load timestamp.*

6.5.2.3. Partitioned Tables

A typical usage of partitioning for manageability is to support a 'rolling window' load process in a data warehouse. The table could be range-partitioned so that each partition contains one month's data. The load process is simply the addition of a new partition. Adding a single partition is much more efficient than modifying the entire table. In practice, one can apply this theory by studying the effects of the load process on a truncated table. Generally one will truncate a table (alternative for a delete) before a data load to avoid data load duplications. According to Nanda (2006), a truncated table in the process of being populated with a load process, is unavailable for general use from the time it is truncated until it is finally loaded. On some very large systems—depending on the complexity of the query, the size of the table, and the general load on both the source and target databases—this process can take hours, during which the users cannot see even the old data (which has been deleted prior to loading). If the INSERT statement fails, due to lack of space or data errors, the users will have to wait until the new data is loaded, which again can be hours. The use of partitions eliminates or largely mitigates these two issues.

Using this piece of advice, ranged partitioned tables were used for the HRDM. Hobbs et al. (2005:115) states that range partitioning is most frequently used where data is partitioned into non-overlapping ranges of data. The column EXTRACT_DATE were used as the partition key, and according to Hobbs et al. (2005:115), the value determines the partition into which a row of data will be placed. This is called "*partition pruning*".

During the staging phase implementation all extracted text files were loaded making use of temporary tables. These temporary tables where then swapped with an empty partition of the relevant ranged partitioned staging table. This meant that any new data could be inserted into any staging table in less than ***0.15*** seconds. This data load technique can also be referred to as "*partitioning swapping*".

6.5.2.4. Dealing with Slowly Changing Dimensions

Asymmetric Hierarchy

One of the biggest challenges with the HRDM was the monthly changes to the organizational hierarchy. A hierarchy contains several levels linked together in a child-parent relationship to form a tree-like structure. The FSDOH organizational structure is represented as an ***asymmetric hierarchy***. According to Malinowski and Zimanyi (2006) an ***asymmetric hierarchy*** has only one path at the schema level but, as implied by cardinalities some lower levels of the hierarchy are not mandatory. In other words, at the instance level the members form a tree where all the branches are not the same length. The asymmetric hierarchy is also of variable-depth due the different branch lengths present. Figure 6-5 shows an example of the FSDOH organizational structure.

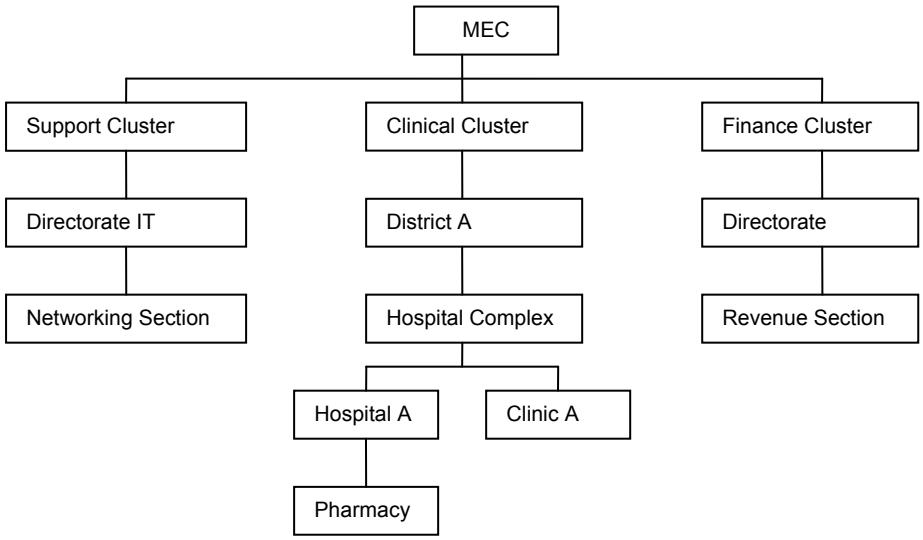


Figure 6-5: FSDOH asymmetric hierarchy (examples of instances)

In terms of cardinality, Malinowski and Zimanyi (2006) states that for both *asymmetric* and *symmetric hierarchies* as implied by the cardinalities, all parent members must have at least one child member and a child member cannot belong to more than one parent member.

Dealing with Changes

Changes occur when new components (organizational units) are created, moved or become obsolete during the month. Components contain the posts for that particular unit and the links of the child components directly reporting to it. Changes in the organizational structure were not directly reflected in each month's download and had to be identified with specially developed algorithms in order to perform SCD Type 2. The complete logical flow of the SCD Type 2 algorithm can be depicted in figure 6-6 using data from July 2007 as an example. Thereafter each step of the general algorithm will be explained in detail.

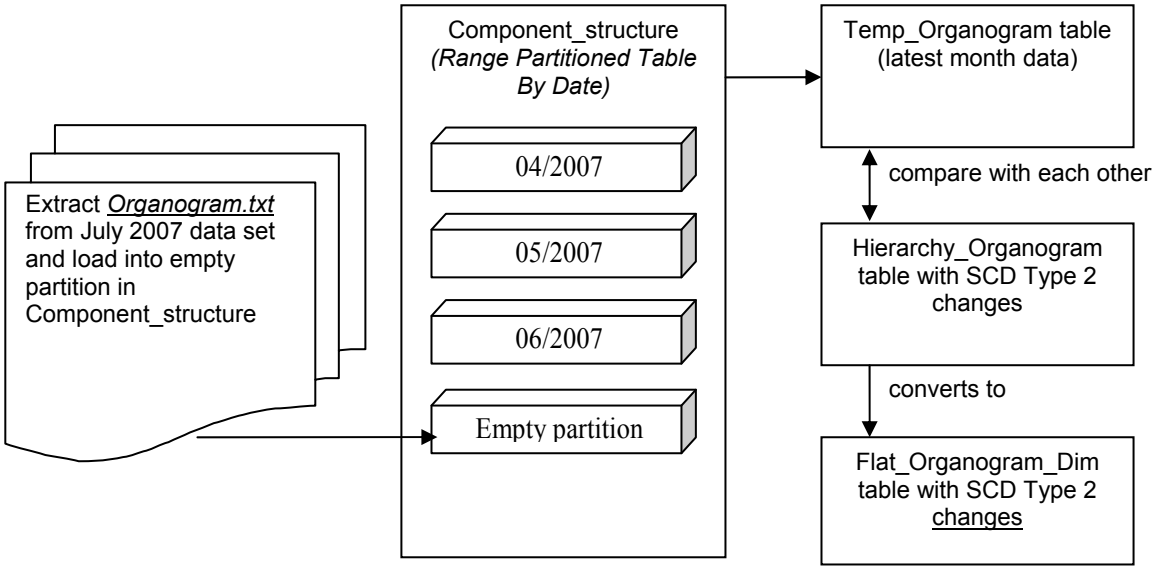


Figure 6-6: Data Flow for Extracted Organogram.txt (June 2007)

Step 1) Load the data snapshot picture

Load the *data snapshot picture* called **Organogram.txt** of the organizational structure into the staging table COMPONENT_STRUCTURE (see table 6-1), with a range partition created on Extract Date. This partitioned table contained the complete organizational hierarchy for each month as was loaded from the data snapshot download. No data changes are performed on this staging table.

Table 6-1: Table Definition of COMPONENT_STRUCTURE

Extract Date	Component	Parent Component	Component Type
01-APR-2007	120382	120380	0015
01-MAY-2007	120382	120380	0015

Step 2) Populate a temporary staging table TEMP_ORGANOGRAM with the latest month data from COMPONENT_STRUCTURE

See Appendix A for the SQL code used to extract the latest month data from the relevant range partition in COMPONENT_STRUCTURE. This selected data was then inserted into the table TEMP_ORGANOGRAM. Important fields to notice are *SCBP* and *COMPONENT_LEVEL*. The *SCBP* field is constructed with the Oracle' SQL engine by concatenating the entire component's hierarchy into a single string. This is a very useful field to detect a level or position change in the organizational structure. Listed below is an example of the *SCBP* entry for the component 120382: **/000004/120000/120376/120377/120378/120380/120382.** *COMPONENT_LEVEL* provides a numeric value which equals the level of the component within the hierarchy.

The remaining portion of the algorithm discussion will concentrate on detecting the changes by algorithmically comparing two database tables using complex SQL statements (Oracle 'CONNECT BY' SQL extension) and SET operators to help identify the following changes:

(1) New Component
(2) Component name change
(3) Component level change
(4) Parent component position change
(5) Parent component name change
(6) Deleted or "missing" Component

Step 3) Detect components that had a position, level or parent component name change. The component and its child components all need to have a flag set to TRUE to make the changes in the hierarchical tree structure.

With the latest month data inserted into TEMP_ORGANOGRAM, the HIERARCHY_ORGANOGRAM and TEMP_ORGANOGRAM tables could be compared with each other to detect changes. It is important to note that both table's data definitions are exactly the same which is a requirement for using Oracle's SET operators. To improve the efficiency of the algorithm, data in HIERARCHY_ORGANOGRAM was extracted in such a way, that the *latest* or *current* version of the hierarchy was used and compared with TEMP_ORGANOGRAM. Thornthwaite (2000) demonstrated the usage of date ranges to obtain a *current* version of a SCD table and the algorithm was adopted and refined to detect HRDM hierarchy SCD2 changes. The modified SQL code *(in italics)* looked like this and is available in Appendix B:

```
WHERE (a.EFFECTIVE_DATE_FROM <= i.EXTRACT_DATE)
AND    (a.EFFECTIVE_DATE_TO  >= i.EXTRACT_DATE OR a.EFFECTIVE_DATE_TO IS NULL)
```

For example, if July 2007 is used as the extract month, the modified SQL code would look like this:

WHERE (a.EFFECTIVE_DATE_FROM <= '01-JULY-2007)
AND (a.EFFECTIVE_DATE_TO >= '01-JULY-2007 *OR a.EFFECTIVE_DATE_TO IS NULL)*

The HRDM algorithm used the above mentioned WHERE clause together with the following fields to detect a change between HIERARCHY_ORGANOGRAM and TEMP_ORGANOGRAM tables:

(1) component number (COMPONENT)
(2) component hierarchy level (COMPONENT_LEVEL)
(3) control component name (CONTROL_COMPONENT_NAME)
(4) component hierarchical position (SCBP)

See Appendix B for the complete SQL code to compare the data in the two tables. When the algorithm detected a change, a new record was INSERTED into HIERARCHY_ORGANOGRAM while the superseding record was changed (UPDATED) to the last date of the previous month. This is the basic requirement when implementing SCD Type 2 changes. See table 6-2 below for a simplified version of HIERARCHY_ORGANOGRAM with component 120382 as an example:

Table 6-2: Table Definition of HIERARCHY_ORGANOGRAM

Extract Date	Component	Component Name	Parent Component	Parent Component Name	Date From	Date To	Parent Change
01-APR-2005	120382	Information Systems	120380	IT (Health)	01-APR-2007	NULL	N

Step 4) Introduce the bridge table FLAT_ORGANOGRAM_DIM

Kimball and Ross (2002:162) pointed out that a hierarchical structure of *variable-depth* presents several problems in the relational environment. According to Corr (2001), these hierarchical structures of *variable-depth* are most often represented in relational databases as **recursive relationships** sometimes known as 'pigs ears' or 'fish hooks'. Some examples of the problems pointed out are the difficulty of navigation or rolling up of facts within these hierarchies using non-procedural SQL. The 'GROUP BY' function in SQL cannot follow the recursive tree structure downward to summarize an additive fact in a companion fact table such as revenue in an organization (Kimball and Ross, 2002:162).

This posed a problem for the FSDOH when using Oracle's 'CONNECT BY' SQL extension to walk each tree node in the same statement as a join. While 'CONNECT BY' is very useful when navigating recursive points in a dimension table, it can not be used by an ad hoc query tool (Corr, 2001). If the tool could generate this syntax to explore the recursive relationship, it can not in the same statement join to a fact table (Corr, 2001). Even if Oracle was to remove this somewhat arbitrary limitation, the

performance at query time would probably be not too good (Corr, 2001). Kimball and Ross (2002:162) also pointed out that the ORACLE CONNECT BY phrase cannot be used in the same SQL statement as a join, which in theory prohibits us from connecting a recursive dimension table to any fact table.

To overcome this problem, a *bridge* table or often called a helper table (Kimball, 1998c) is inserted between the hierarchical dimension table and the fact table (Kimball and Ross, 2002:163). The problem the FSDOH experienced with this approach was entrenched in the manner the OLAP tool used the dimensional model for its analytical functions (*roll-up* and drill-down) on the organization dimension. Malinowski and Zimanyi (2004) also points out that OLAP tools usually only cope with hierarchies that ensure summarizability or that can be transformed so that summarizability conditions hold. Summarizability refers to the correct aggregation of measures in a higher hierarchy level taking into account existing aggregations in a lower hierarchy level (Malinowski and Zimanyi, 2004). Therefore the OLAP tool required a flattened view of the hierarchical organizational structure to function appropriately. The newly created flattened view will contain the child-parent relationships in HIERARCHY_ORGANOGRAM (see figure 6-7) in terms of a column for each tree level. The table FLAT_ORGANOGRAM_DIM was created as a ***modified version of a bridge table*** and used as one of the dimensions in the dimensional model.

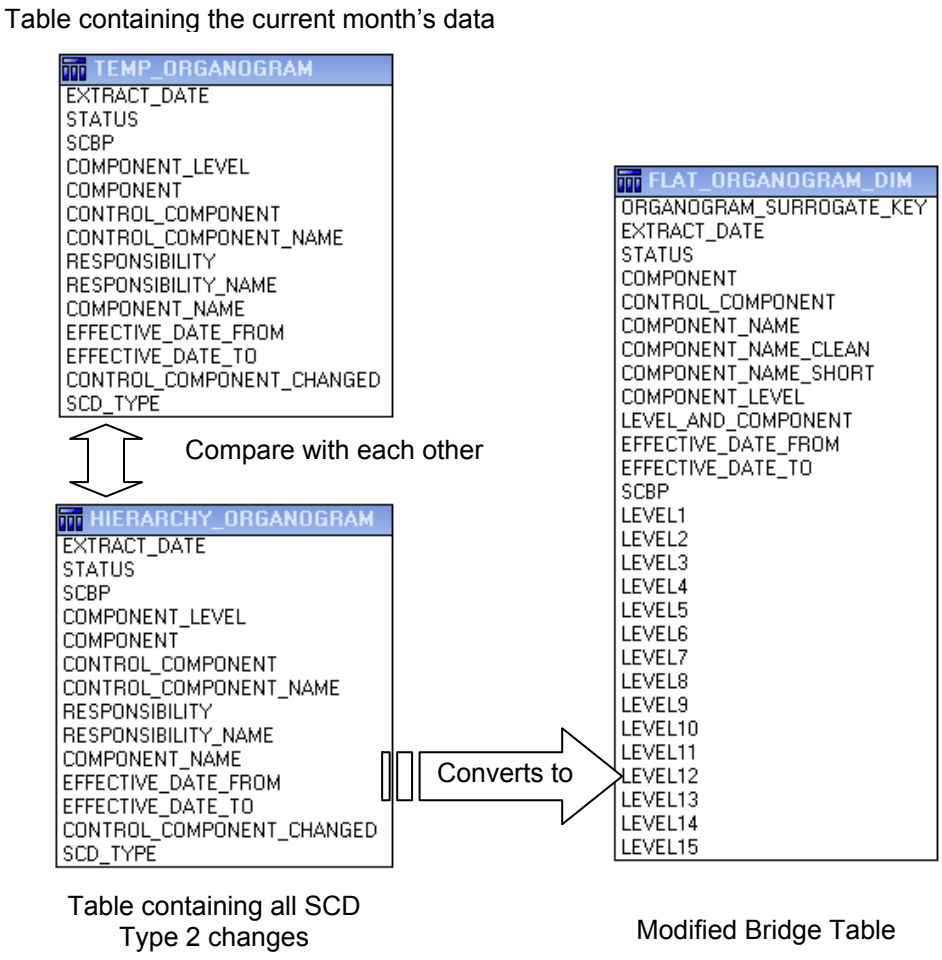


Figure 6-7: Portion of the Dimensional Model

A surrogate key called ORGANOGRAM_SURROGATE_KEY (see below) was created as the FLAT_ORGANOGRAM_DIM table's primary key. It contained the extract date (*load timestamp*) concatenated with the component number to stay within the bounds of SCD Type 2 changes.

(Rule): ORGANOGRAM_SURROGATE_KEY=TO_DATE (EXTRACT_DATE,'dd-MON-yyyy')||'-'||COMPONENT

(Example): ORGANOGRAM_SURROGATE_KEY (**011324**) = **'01-JUL-2007-011324'**

According to Kimball (1998a) a surrogate key is just anonymous integers. This approach differs from the traditional Kimball view of surrogate keys, as it would just add unnecessary columns to this table. Instead of an anonymous integer, ORGANOGRAM_SURROGATE_KEY was constructed with meaning but at the same time, also provided a unique primary key for the FLAT_ORGANOGRAM_DIM table. See Appendix C for the SQL code to construct the FLAT_ORGANOGRAM_DIM table.

Step 5) Construct the core Dimensional Model

Dimension Tables

With the modified bridge table FLAT_ORGANOGRAM_DIM in place, the last step was to construct the dimensional model. A conglomerated table called SCD_TYPE2_BRIDGE (see figure 6-8) was constructed. This table joined selected columns from staging tables POSTS and PERSONNEL together. The purpose of this conglomerated table was to provide a platform for ensuring all posts and personnel row entries were joined successfully together. The total row count of this join must equal the row count of posts. Data quality routines were dependent on the SCD_TYPE2_BRIDGE table. This table will also form the basis of constructing the HR_FACT fact table later on.

Next, the dimension tables HR_PERSONNEL_DETAILS_DIM and HR_POST_DETAILS_DIM were derived from the SCD_TYPE2_BRIDGE table respectively. In other words SCD_TYPE2_BRIDGE = HR_PERSONNEL_DETAILS_DIM + HR_POST_DETAILS_DIM. The cardinality between SCD_TYPE2_BRIDGE → HR_PERSONNEL_DETAILS_DIM can be expressed as (1) → (1) and SCD_TYPE2_BRIDGE → HR_POST_DETAILS_DIM also as (1) → (1). SCD_TYPE2_BRIDGE's main purpose was to reduce SQL join time and provide a stable platform to be able to perform data quality routines.

Surrogate keys played an important part in constructing the dimensional model and two surrogate keys named PERSONNEL_SURROGATE_KEY and POST_SURROGATE_KEY were introduced into SCD_TYPE2_BRIDGE. The surrogate keys (PERSONNEL_SURROGATE_KEY and POST_SURROGATE_KEY) were constructed using a concatenated combination of the following columns in the staging tables POSTS and PERSONNEL:

- EXTRACT_DATE (staff details)
- PERSAL_NUMBER (staff details)
- POST_NUMBER (post details)
- POST_TITLE (post details)
- COMPONENT (post details)

Depicted below in figure 6-8 is the SCD_TYPE2_BRIDGE table with the fact and dimension tables that will be extracted from it. This extraction phase of the fact table will now be discussed.

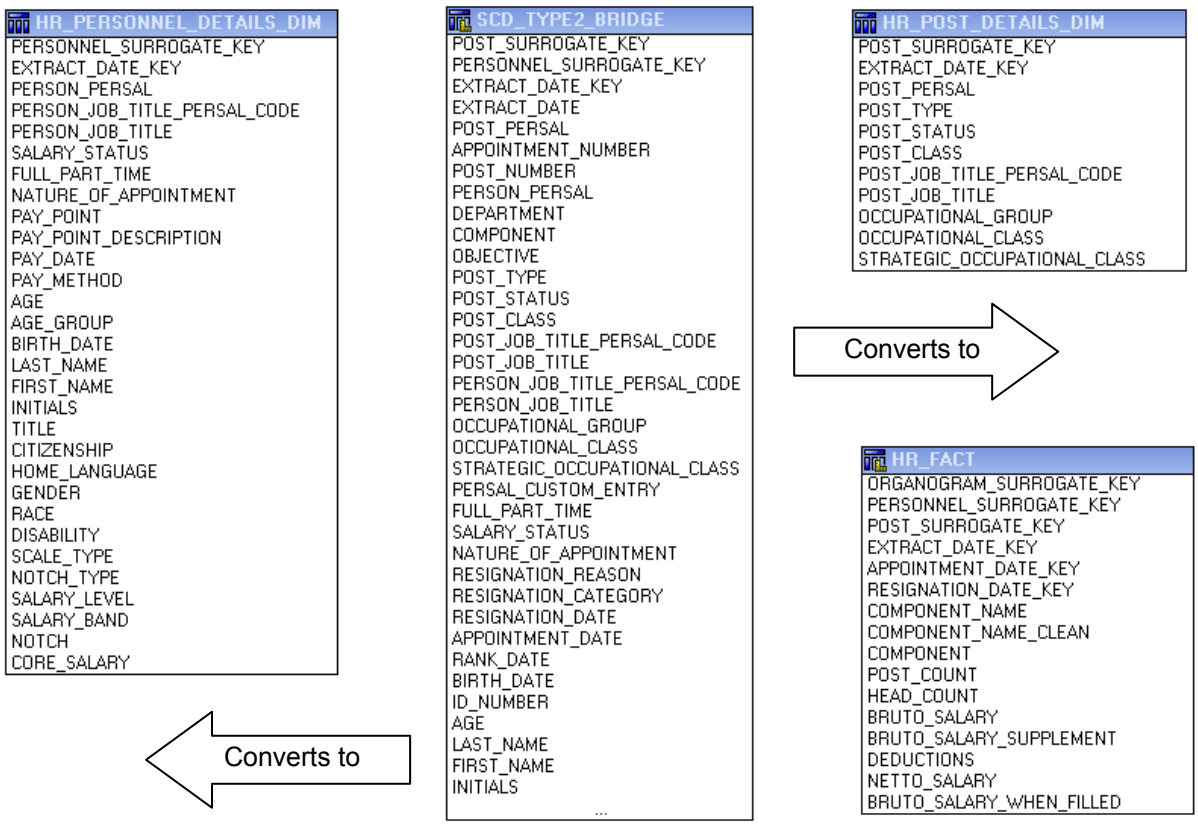


Figure 6-8: SCD_TYPE2_BRIDGE table

Fact Table

With the dimension tables in place, the next step was to construct the necessary fact table in the dimensional model. According to Becker (2004), one of the problems of using the SCD Type 2 technique is the large number of additional rows required to support all the changes. Barbusinski, Howard, Jennings, Kelley and Oates (2003) pointed out that joining the fact and associated dimensions would also require complex temporal joins at analysis time. Furthermore the SQL statement must include time reference for both the fact and associated dimensions. All these factors will lead to an undesired environment for non-sophisticated users such as is the case of the FSDOH.

One possible way of overcoming these obstacles was by using a materialized view (mview) to hide the design complexity. According to Goldstein and Larson (2001) query processing time can be improved through the use of materialized views. Taking the advice of Goldstein and Larson, a materialized view called HR_FACT (see figure 6-8) was created between the joins of the tables FLAT_ORGANOGRAM_DIM and SCD_TYPE2_BRIDGE as can be seen in the SQL code below.

```
CREATE MATERIALIZED VIEW HR_FACT
PCTFREE 0
NOLOGGING
BUILD IMMEDIATE
REFRESH COMPLETE ON DEMAND
AS
SELECT
   a.ORGANOGRAM_SURROGATE_KEY
  , b.PERSONNEL_SURROGATE_KEY
  , b.POST_SURROGATE_KEY
  , b.EXTRACT_DATE_KEY
  , a.COMPONENT
  , a.COMPONENT_NAME
  , a.COMPONENT_NAME_CLEAN
  , b.POST_COUNT
  , b.HEAD_COUNT
  , b.BRUTO_SALARY
  , b.BRUTO_SALARY_SUPPLEMENT
  , b.DEDUCTIONS
  , b.NETTO_SALARY
  , b.BRUTO_SALARY_WHEN_FILLED
FROM flat_organogram_dim A, scd_type2_bridge b
WHERE a.COMPONENT = B.COMPONENT
AND a.EFFECTIVE_DATE_FROM <= b.EXTRACT_DATE
AND (a.EFFECTIVE_DATE_TO >= b.EXTRACT_DATE OR a.EFFECTIVE_DATE_TO IS NULL)
```

The SQL join section (WHERE … clause) provided the mechanism of selecting the correct surrogate key (ORGANOGRAM_SURROGATE_KEY) from FLAT_ORGANOGRAM_DIM to be inserted into the fact table HR_FACT. This was done by joining the rows in SCD_TYPE2_BRIDGE with the relevant hierarchy entity that also matches the 'type 2' change in that ***time period***. This portion also successfully "wrapped" the complexities of all the SCD Type 2 joins in the SQL statement and created an environment that is easy to use for user navigation. HR_FACT was introduced as the main fact table into the dimensional model.

Step 6) Expand the core Dimensional Model to accommodate additional subject areas

Just having facts on the number of posts and personnel was only addressing the first two business requirements of section 6.5.1. Metrics on absenteeism and qualifications were absent in the core dimensional model. In order to provide these additional metrics, the data mart had to be expanded to include the ABSENTEEISM and QUALIFICATIONS sub-subject areas within the human resources subject area. Both these sub-subject areas will now be discussed in terms of ETL and the construction of their respective dimensional models.

ABSENTEEISM SUB-SUBJECT AREA

The data for this subject area was loaded into the LEAVE_TAKEN staging table (see figure 6-9). An interesting challenge presented itself with this particular sub-subject area. The data in this staging table was not on *the day level* grain and instead it was in the form of a BEGIN_DATE to END_DATE, for example "01-APR-2007" to "10-APR-2007". In order to perform analytics a record is required for each single date for example "01-APR-2007", "02-APR-2007", "03-APR-2007" up to "10-APR-2007". ***In total 10 records or fact entries are required*** if the begin date and end date is respectively 01-APR-2007 and 10-APR-2007. A SQL algorithm was developed to perform the necessary conversation and used a FOR-LOOP to break the data range to individual days. The converted data was then loaded into a 2nd level staging table called LEAVE_TAKEN_ACROSS_DAYS (see figure 6-9).

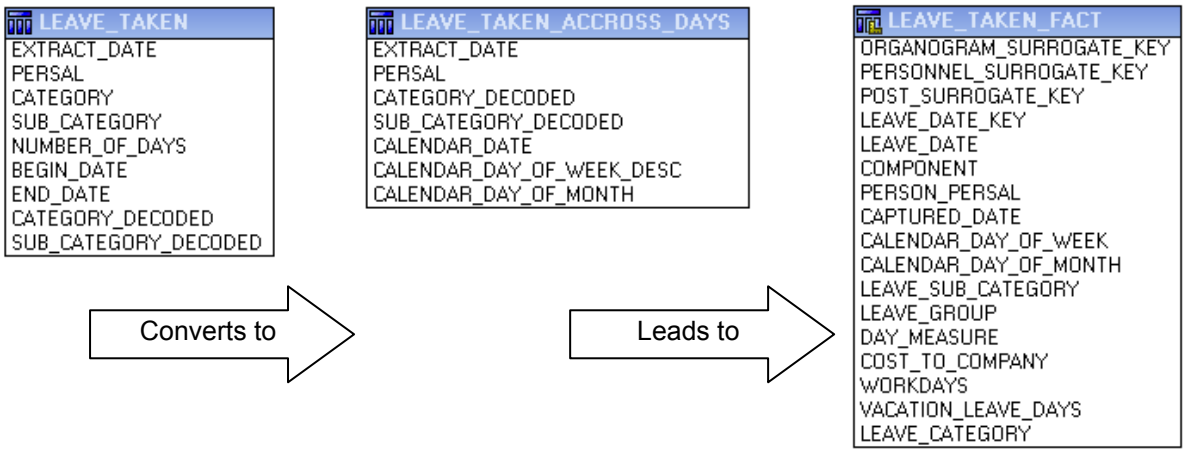


Figure 6-9: Absenteeism Sub-Subject Area (Leave Taken)

The next step was to create a fact table called LEAVE_TAKEN_FACT (see figure 6-9) from this 2nd level staging table. In an earlier chapter the concept of a consolidated fact table was introduced. It was pointed out that consolidated fact tables can assist if the requirement of measuring more than one metric with each other within the same fact table exists. The business requirement was to calculate the ***absenteeism rate***. In order to calculate the absenteeism rate, the numerator would be the "amount of days on leave" and the enumerator would be "working days in a month": Absenteeism rate = total days leave taken / working days. The fact table was created as a

consolidated fact table but introduced the issue of "double counting". Knutsen (2005:23) mentioned some principles of solving 'double counting' of facts and our algorithm employed ***method 2*** ("add rows to the fact table for each fact, using zeros in the unused fact columns"). Tabulated below (see table 6-3) is an extract for an individual using only important columns and included data from April 2007 until August 2007 to illustrate how ***method 2*** was used.

Table 6-3: Portion of LEAVE_TAKEN_FACT

Leave_date_key	Persal	Day_Of_week	Day Measure	Cost to Company	Workdays
20070504	10208135	FRIDAY	1	379.50	0
20070503	10208135	THURSDAY	1	379.50	0
20070613	10208135	WEDNESDAY	1	379.50	0
20070612	10208135	TUESDAY	1	379.50	0
20070731	10208135	TUESDAY	1	407.86	0
20070810	10208135	FRIDAY	1	407.86	0
20070919	10208135	WEDNESDAY	1	407.86	0
20070401	10208135		0	0	18
20070501	10208135		0	0	22
20070601	10208135		0	0	21
20070701	10208135		0	0	22
20070801	10208135		0	0	22

With all the leave taken data available the only outstanding data was leave credits. The term "Leave credits" refers to the number of leave days allocated to a person per year, and how many days are still remaining for the months after a leave cycle starts. For the FSDOH a leave cycle is twelve (12) months. Data was loaded into the LEAVE_CREDITS staging table and the fact table called LEAVE_CREDITS_FACT was populated by a simple extract from the staging table (see figure 6-10).

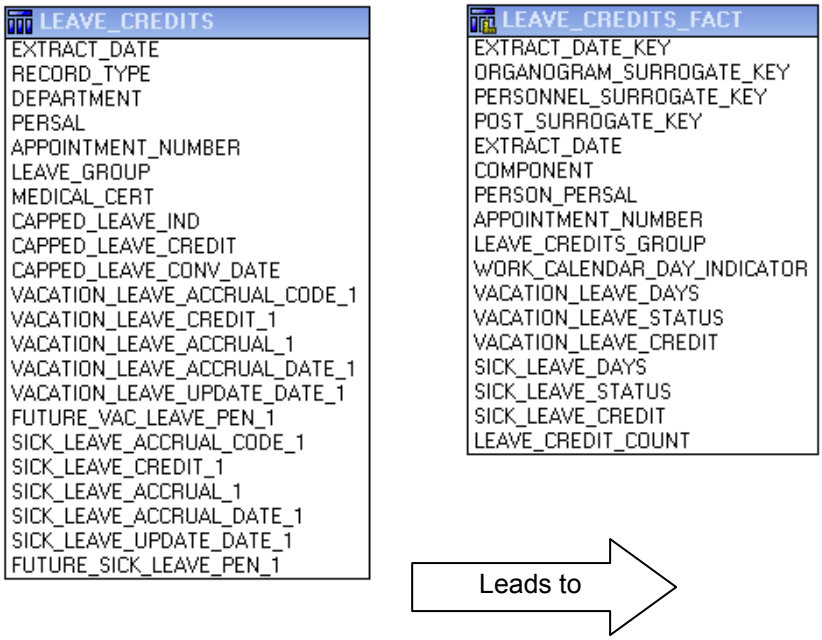


Figure 6-10: Absenteeism Sub-Subject Area (Leave Credits)

QUALIFICATIONS SUB-SUBJECT AREA

The data for this sub-subject area was loaded into the QUALIFICATIONS staging table (see figure 6-11). The data in the QUALIFICATIONS table was easy to interpret and did not require special ETL interventions. The fact table QUALIFCATIONS_FACT (see figure 6-11) was constructed using only the necessary columns of the QUALIFICATIONS and introduced into the HR data mart and dimensional model.

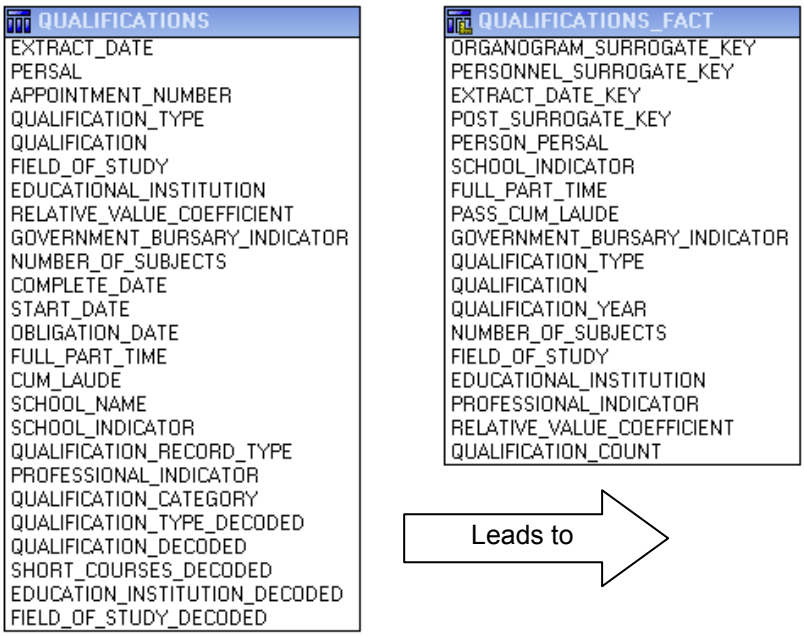


Figure 6-11: Qualifications Sub-Subject Area

Step 7) Finalize the Dimensional Model

The final dimensional model consisted of the following fact and dimension tables and is illustrated on the following page (see figure 6-12). The necessary indexes were created on the respective columns, to assist in speeding up the necessary SQL-JOINS.

The tables LEAVE_DATES_DIM and FINANCIAL_DATES_DIM are *role-playing dimensions* and are constructed using the conformed dimension DATE_DIM table. In practice this meant that data was selected from the DATE_DIM table for the *role-playing dimensions* when *SQL joins* took place with the fact table. Only selected fields were used for the *role-playing* dimensions in the HRDM dimensional model as can be seen in figure 6-12.

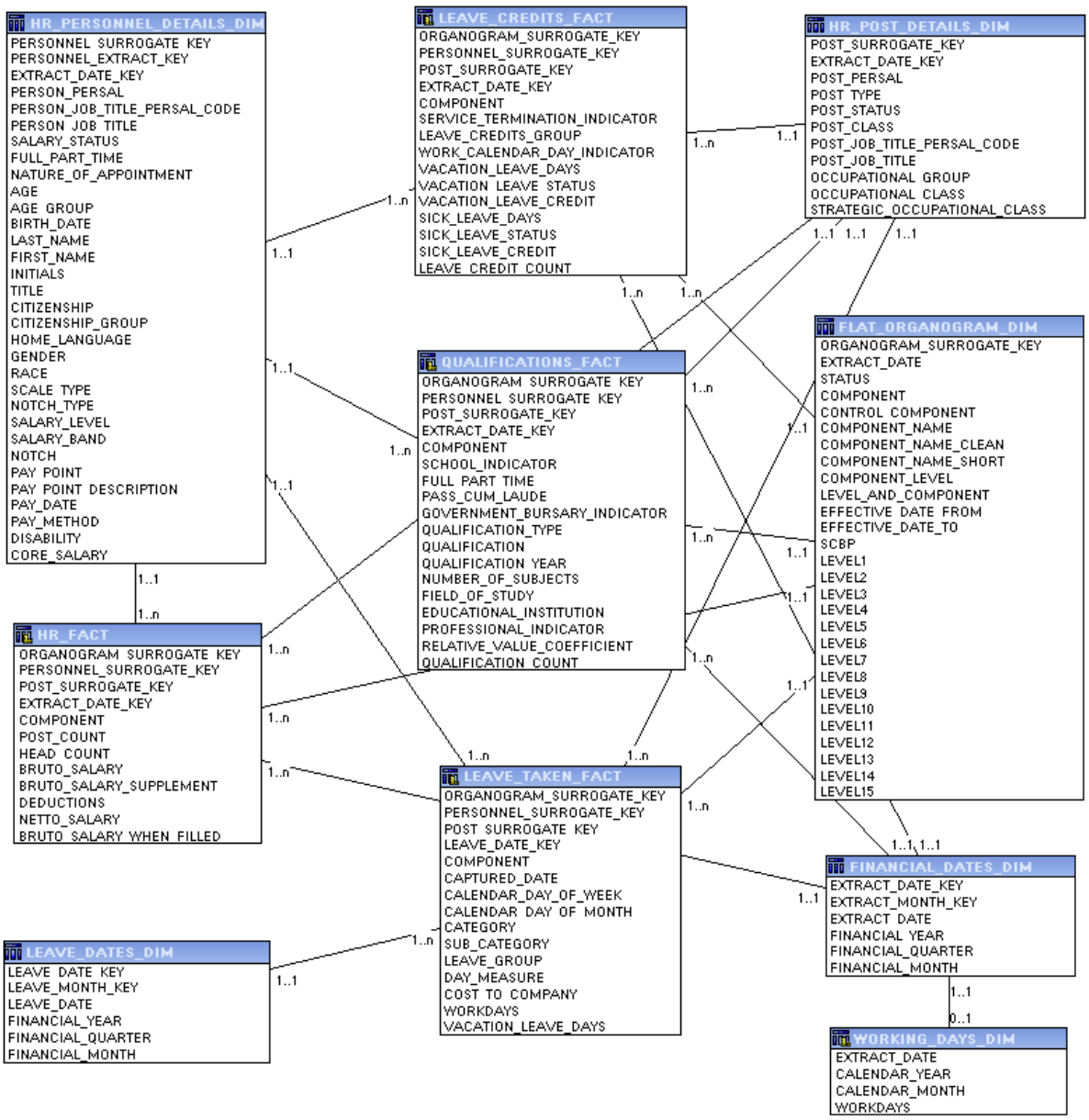


Figure 6-12: HRDM Dimensional Model

6.5.3. OLAP Dimensional Model

An important aspect that received attention was to provide an analytical environment by means of an OLAP cube. A denormalized materialized view called HR_FACT_MOLAP_FLAT (see figure 6-13) was created and used in Cognos Transformer (OLAP tool) to construct the HR cube. This materialized view joined HR_FACT, FLAT_ORGANOGRAM_DIM and SCD_TYPE2_BRIDGE (HR_PERSONNEL_DETAILS_DIM, HR_POST_DETAILS_DIM) together to produce the data needed for the OLAP building tool.

Similarly the materialized views LEAVE_TAKEN_MOLAP_FLAT, TAKEN_CREDITS_FACT_MOLAP_FLAT and QUALIFICATIONS_FACT_MOLAP_FLAT were constructed and used to respectively create cubes for ABSENTEEISM and QUALIFICATIONS.

The dimensions FLAT_ORGANOGRAM_DIM, FINANCIAL_YEAR_DIM and LEAVE_DATES_DIM were reused and introduced into the OLAP dimensional model. The final OLAP dimensional model consisted of the following fact and dimension tables (see figure 6-13).

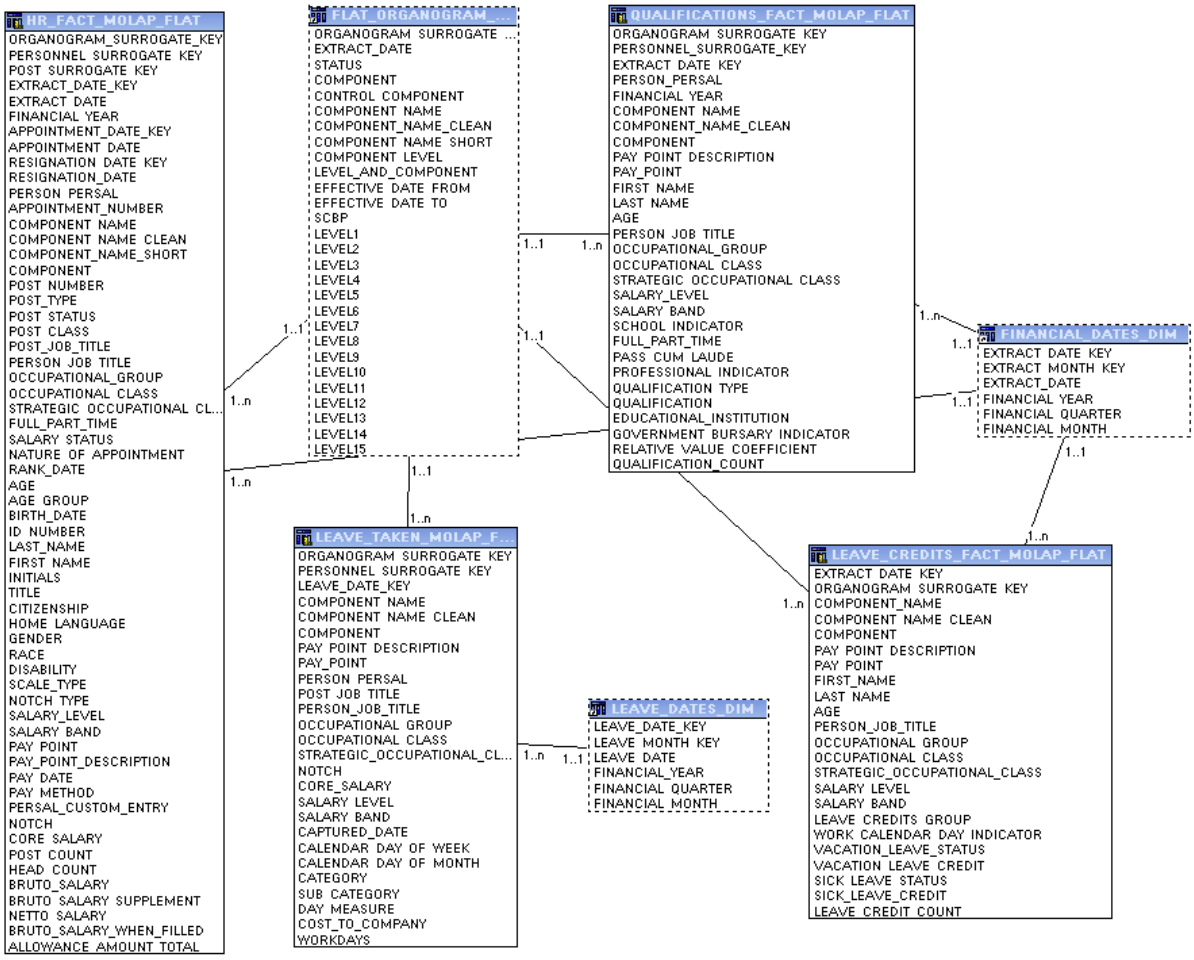


Figure 6-13: Complete HRDM OLAP Dimensional Model

6.6. Development of the ARV Data Mart (ARVDM)

As mentioned in Chapter 3, the MRC in collaboration with the UCT Lung Institute developed a standalone ARV data warehouse for monitoring and evaluation of the Free State ARV Treatment Programme. The usage and design of this data warehouse focused on ***clinical outcomes for the ARV research initiative.*** Due to the fact that the ARV data warehouse was also situated in Cape Town, a frustrated ARV Management team in the Free State had no opportunity to interface with the data warehouse directly. Only static reports were available and they were designed to answer the clinical treatment outcomes needed for the National Indicator Report, which was required by the National Department of Health. No *internal required* strategic information was extracted to support the decision-making or planning processes of the ARV Management Team in the FSDOH

In March 2006, a critical decision was made by the FSDOH executive team which changed the public face of ARV treatment in the Province. The ARV treatment programme would no longer operate as an isolated vertical project but would instead form part of the Integrated Health Care Plan (IHCP). *This strengthened the need to integrate the ARV data warehouse into the FSDOH data warehouse.* In June 2006 an executive-level decision was taken by the FSDOH management team to hand-over the data warehousing project to the *author of this thesis* and to integrate the ARV Data Warehouse into the FSDOH Data Warehouse as a separate ARV data mart (ARVDM). This section will cover the integration of the ARVDM and the managerial outcome requirements.

6.6.1. Addressing Managerial Outcomes

An ARV data warehouse can assist management with determining the breakdown of costs associated with each element of the ARV treatment programme. The FSDOH managerial outcomes differ from previous studies and the following outcomes have been identified to be monitored and evaluated:

External Management Requirements:

- Reports on the Division of Revenue Act (DORA) Indicators to be used for funding the ART Programme from the Conditional Grants provided by National Treasury.
- Annual and Quarterly Reports on National Department of Health (DoH) Indicators
    - Pharmaceutical (Regimen) Indicators
    - Clinical Treatment Outcome Indicators (ART > 200, WHO Staging, Regimen1a, Regimen2b, Regimen2 breakdown)

Internal Management Requirements:

- *Clinical Treatment Outcome* – Determine the success rate of patients on the ARV Treatment Programme. Can also be referred to as the antiretroviral therapy adherence indicators.

6.6.2. Extraction, Transformation and Loading Challenges

The ETL process made use of a staging area which is hosted on a SQL*Server database and a data mart which is hosted in the FSDOH data warehouse. All of the data (14 text source) files are extracted from Meditech on a weekly basis and each file data elements match a CDS (Customer Defined Screen) on the MPM application. These CDS screens in turn were based on the paper-based information system which was also discussed in Chapter 3. These text files are loaded into the SQL*Server staging database. A standalone dimensional model was created within the SQL*Server staging database. This dimensional model was then mirrored into the Oracle data warehouse and additional fields and tables were added. These tables and fields were required to adopt the dimensional model to fit into the FSDOH data warehouse bus architecture (see figure 6-14).

The *role-playing dimension* ARV_TREATMENT_LOCATION_DIM was added to the dimensional model as a database view of the TREATMENT_LOCATION_DIM dimension table (using the filter ARV_FLAG='TRUE'). All date *role-playing dimensions* were constructed as database views from the DATE_DIM dimension table and introduced as time dimensions.

6.6.2.1. Patient Confidentiality

To ensure patient confidentially, the ARV_PATIENTS_DIM table did not contain any personal identification fields (except gender and date of birth). All sensitive personal identification fields as well as next of kin fields were moved into a separate ARV_PATIENTS_DEMO_DIM table (see figure 6-14). The ARV_PATIENTS_DEMO_DIM table was hidden from users when they interfaced to the data mart using Cognos Query Studio and Cognos Report Studio.

6.6.2.2. Linking with the HRDM

To provide the functionality of linking the ARV fact table with the HRDM, the PERSAL (a unique personnel identifier) number of each clinician or nurse providing a clinical service is captured on each Meditech screen and included in each week's data download. The PERSAL number is transformed into a field called Personnel_Extract_Key which is a combination of the PERSAL number and the 1st day of the month the clinical action was performed. Remember all PERSAL data is downloaded once a month and time stamped with the 1st day of the month (EXTRACT_DATE). This transformed key links up with a matching key in HR_PERSONNEL_DETAILS_DIM. ARV_HR_PERSONNEL_DIM (see figure 6-14) was added as a *role-playing dimension* by creating a database view of HR_PERSONNEL_DETAILS_DIM (see figure 6-12).

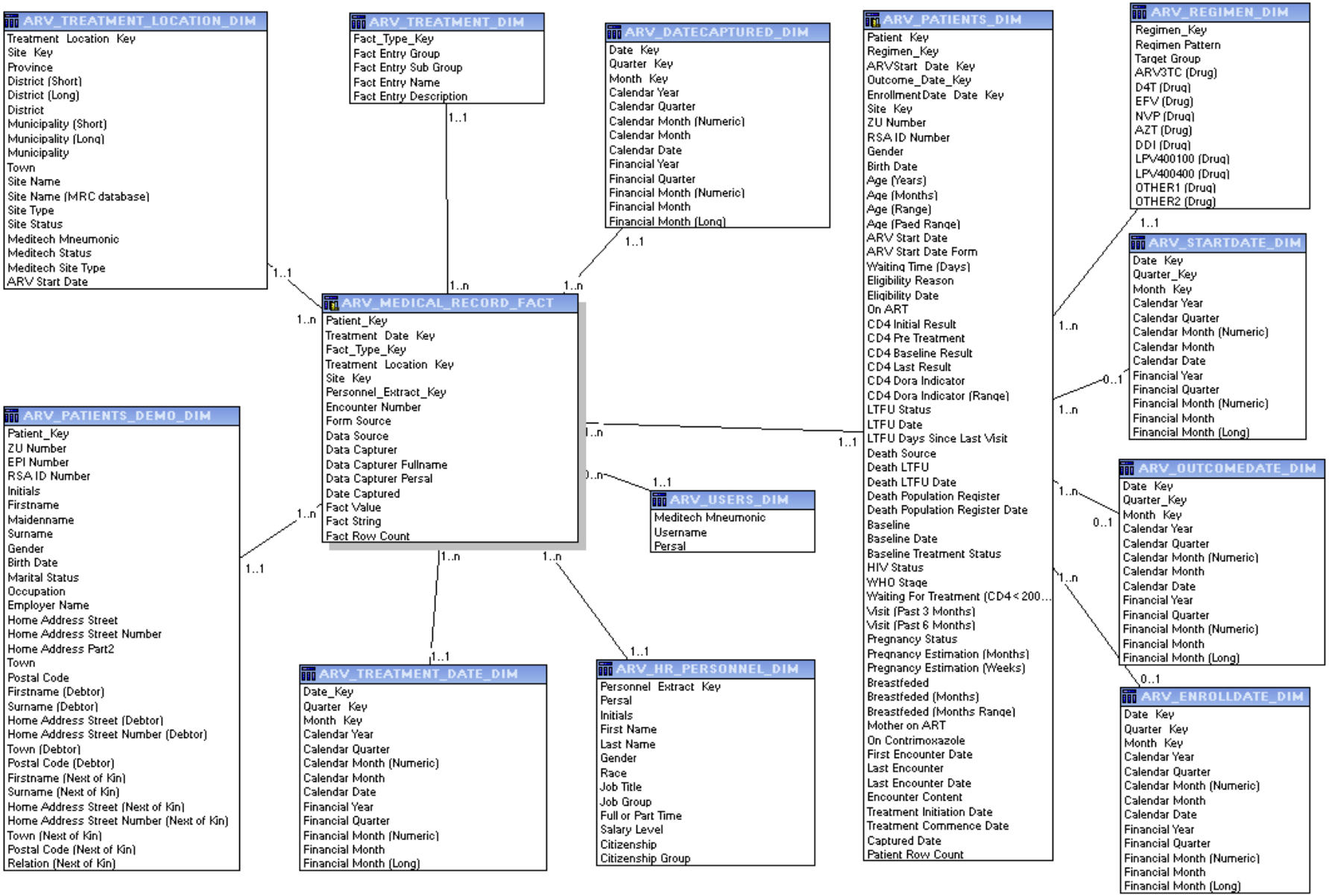


Figure 6-14: ARVDM Dimensional Model

6.7. Abstracting of the ARV Human Resources Data Mart (ARVHRDM)

With the integration of the ARV data warehouse, additional strategic information was identified. The ARV Management team needed strategic information as part of the Division of Revenue Act (DORA) for obtaining human resources funding from National Treasury. In order to answer these questions an abstraction of the HRDM was developed called the ARVHRDM. All dimension and fact tables have been reused in this data mart with the exception of FLAT_ORGANOGRAM_DIM. The HRDM comprises **all** the organizational units in a hierarchical fashion which is contained within the FLAT_ORGANOGRAM_DIM table. The ARV programme required a much simpler hierarchy of and would only required ARV programme organization units (hospitals and clinics). In order to achieve this changed in requirement, the following section will describe the managerial outcomes, ETL challenges and the final dimensional model of the ARV Human Resources Data Mart (ARVHRDM).

6.7.1. Addressing Managerial Outcomes

External Management Requirements:

- Number of ARV posts filled/vacant by DORA occupational classes
- Cost of salaries for ARV posts filled

Internal Management Requirements:

- Absenteeism rate of ARV staff
- Cost associated with filling vacant ARV posts
- Individual ARV absenteeism patterns of individuals
- Number of ARV vacant posts
- Qualifications distribution of staff on the ARV programme

6.7.2. Extraction, Transformation and Loading Challenges

ARV_UNITS_DIM intervention

In order to reuse the dimensional model of the HRDM, a bridge tabled called ARV_UNITS_DIM (see figure 6-15) was constructed as a subset from the FLAT_ORGANOGRAM_DIM table. The ARV_UNITS_DIM table was populated with entries that are only ARV sites (hospitals and clinics). By reusing the HRDM, an enormous amount of time was saved in development time and this data mart could simply just build on the success already achieved.

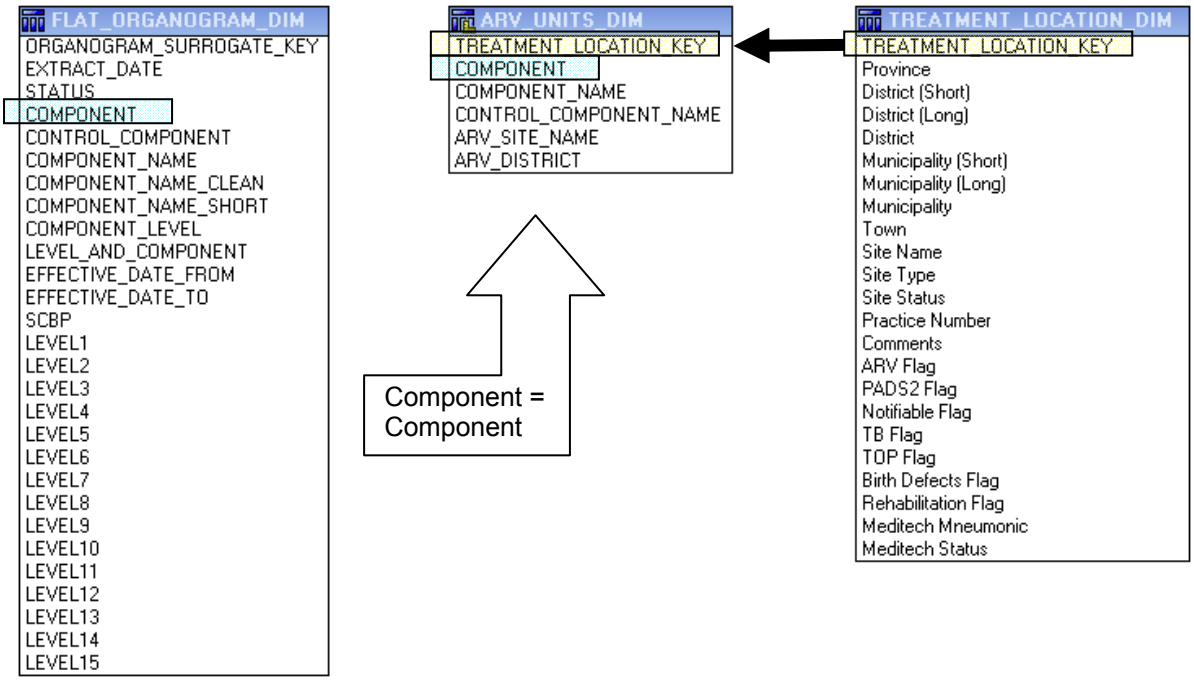


Figure 6-15: Portion of the ARVHRDM Dimensional Model

A column called ARV_SITE_NAME was added to ARV_UNITS_DIM to allow any SQL statement to combine all the relevant components for a selected ARV site under a single ARV_SITE_NAME description. This removed the arbitrary 1 → M cardinality that existed in the FLAT_ORGANOGRAM_DIM and simplified the dimensional hierarchy as can be seen below.

ARV District ⟶ ARV Site Name ⟶ Component Name ⟶ Component

The tables ARV_UNITS_DIM and TREATMENT_LOCATION_DIM were linked with each other, by manually comparing the ARV_SITE_NAME and "Site Name" in TREATMENT_LOCATION_DIM. The correct TREATMENT_LOCATION_KEY field was inserted into the ARV_UNITS_DIM table. Tabulated below (see table 6-4) is an example of "***Bophelong Clinic (Vrede)***" and the associated dimensional data extracted from ARV_UNITS_DIM using this simplified approach.

Table 6-4: Example using ARV_UNITS_DIM

ARV_UNITS_DIM	With Data
TREATMENT_LOCATION_KEY	748
COMPONENT	010834
COMPONENT NAME	**HIV/AIDS COMPRE/MANAGEMENT CENT(BOPHELONG CL(DC19**
ARV SITE NAME	***Bophelong Clinic (Vrede)***
ARV DISTRICT	Thabo Mofutsanyane (DC19)

6.7.3. Dimensional Model

As mentioned before, the dimensional model (see figure 6-16) was derived from the original HRDM. The FLAT_ORGANOGRAM_DIM table was replaced by the ARV_UNITS_DIM table to enable the simplified location lookup. All fact and the dimension table ARV_UNITS_DIM were joined on the COMPONENT field instead of using the ORGANOGRAM_SURROGATE_KEY.

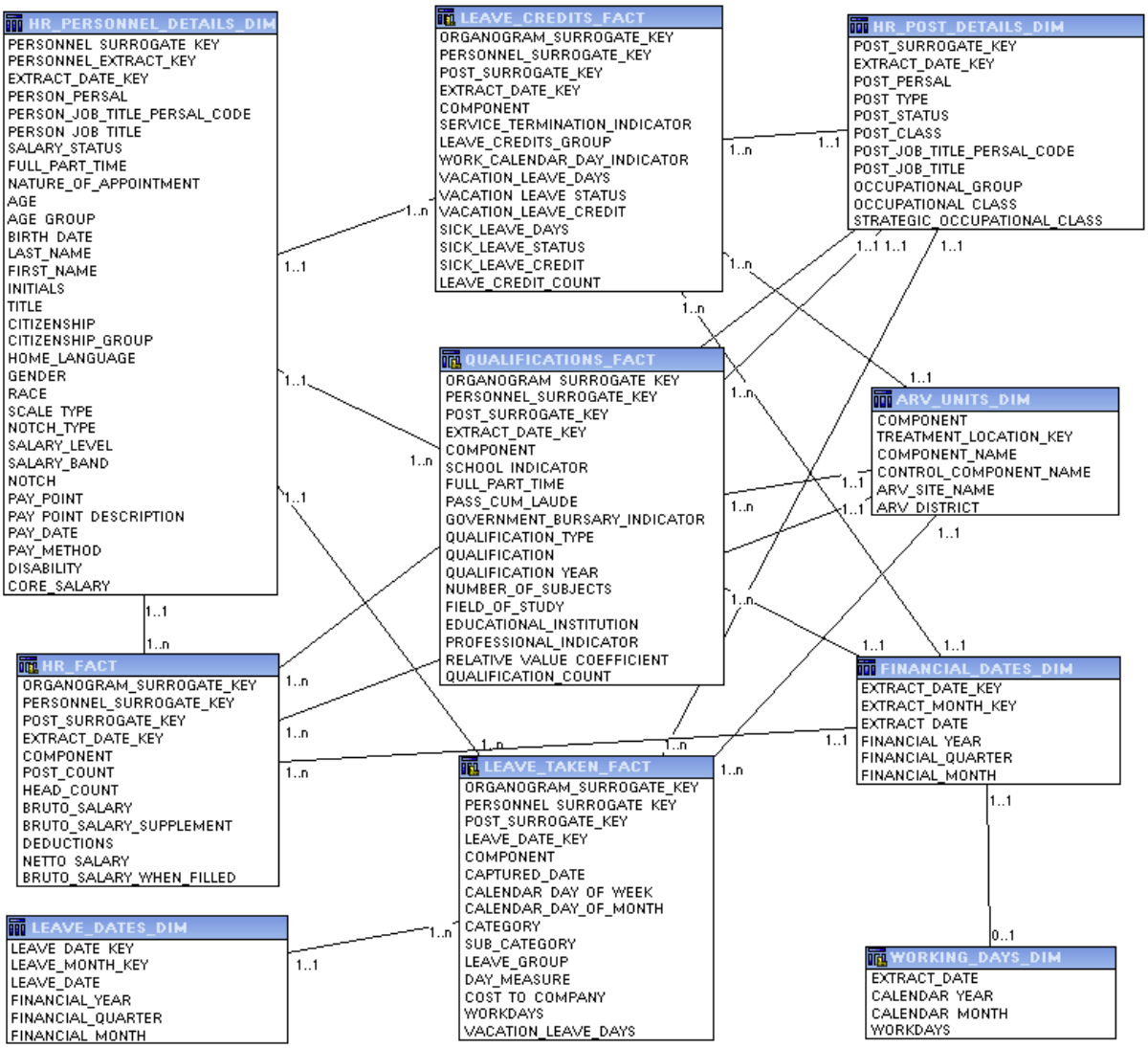


Figure 6-16: ARVHRDM Dimensional Model

6.8. Development of the PADS Data Mart (PADSDM)

The Patient Admission and Debiting System (PADS) was implemented in 29 district and regional hospitals across the Free State province. This online system offers the functionality of a basic HIS and was developed by in-house staff of the FSDOH. The functionality the system covers includes Patient Admission, Discharges, Billing, Payments, Clinical Diagnoses and a Master Patient Index. While Meditech is deployed at the larger institutions and all ARV treatment sites in the Free State, the PADS system is deployed at the remaining hospitals. The reason for this mixed approach can be found in the lack of the necessary financial resources to implement Meditech in at all hospitals in the Free State.

The system was developed in Oracle 10g and has been used in a web-based operational environment since July 2000. To date, almost 2.4 million admissions have been entered on the system and it boasts a Master Patient Index of 1 345 095 unique entries. This data mart's primary goal was to provide strategic information on revenue collection. With the integration of the ARV data warehouse, a secondary goal was identified to gather and link all clinical information of ARV patients that were admitted into hospitals for non-ARV treatment. This clinical information would then be made available in the ARVDM and assist in building a complete clinical picture of a patient on ARV treatment.

6.8.1. Addressing Managerial Outcomes

External Management Requirements:

- Identify patients with an outstanding balance > R200 and older than 90days. These patient accounts must be handed over to external debt collectors
- Levies and payments per medical aid, and submitted to each Medical Aid for verification

Internal Management Requirements:

- Provide clinical information on any identified ARV patient in the PADS database
- Payment types and patterns analyses per institution, per year
- Detailed Listing of "handed over" accounts per institution for each debt collector
- Calculate and provide average length of stay (ALOS)
- Analysis of foreign patients and outstanding balances by these patients

6.8.2. Extraction, Transformation and Loading Challenges

The ETL process was divided into two stages to simplify the entire process. Stage One (1) dealt with populating the base staging tables while Stage Two (2) dealt with constructing the consolidated tables.

6.8.2.1. Stage One

In this stage, the base stage tables (see figure 6-17) were populated from the operational data source. A full refresh technique was implemented due to the volatility of the data in the operational data source. The full refresh process was scheduled for once a week and executed using a Linux operating system (OS) “cronjob” on a Sunday afternoon.

Figure 6-17: PADSDM ETL Stage 1

6.8.2.2. Stage Two

During this stage of the ETL process the combined tables VISIT_DETAILS and FLOW_EVENTS (see figure 6-18) were constructed. These tables served as the foundation for Cognos Transformer, the Cognos OLAP tool used to build cubes. Their primary goal was to reduce the exhausting processing time caused by the expensive SQL-joins during cube creation. VISIT_DETAILS was a SQL-join result between VISITS, ACCOUNTS, DISCHARGES, ADMISSIONS and PATIENTS. FLOW_EVENTS was constructed by inserting table rows data from the tables SERVICES, ACCOUNTS and REVENUE and is classified as a consolidated fact table.

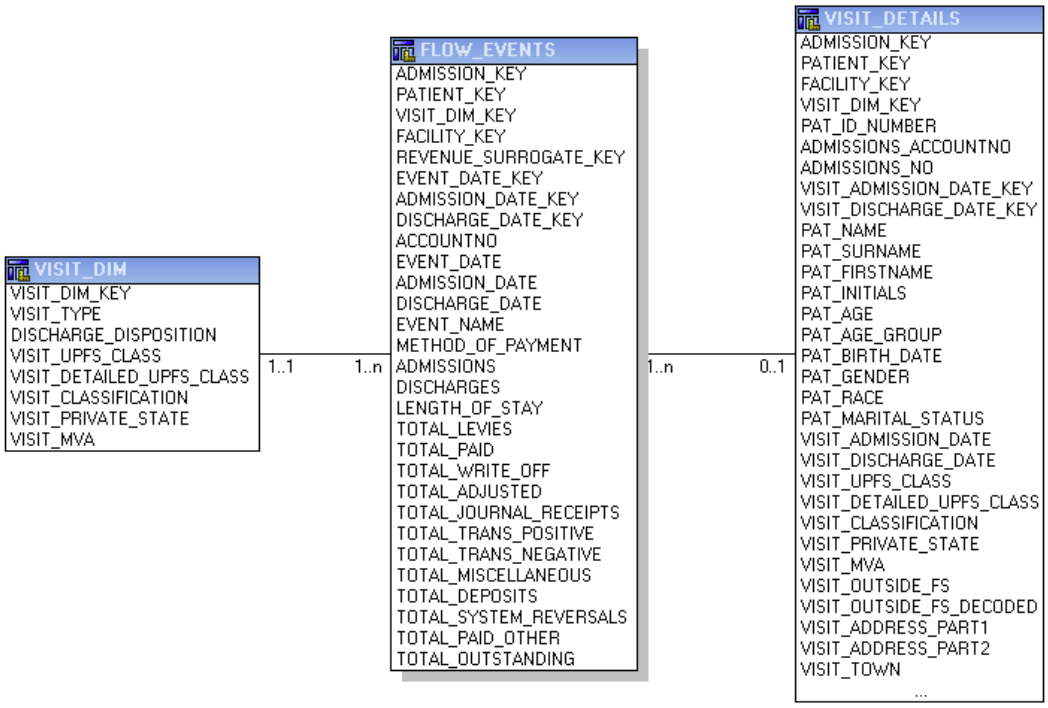


Figure 6-18: PADSDM ETL Stage 2

Construction of VISIT_DETAILS

Alarming performance degradation was observed after the implementation of the initial VISIT_DETAILS table. Users pointed out slow query times when constructing ad-hoc queries using **Cognos Query**. Most of the observed performance degradations involved filtering an attribute and linking it to the fact table. By avoiding a separate lookup dimension table in the dimensional model design, the database scanned all 2,067,508 rows to obtain a selection set from VISIT_DETAILS. In an attempt to overcome this limiting factor, a bridge VISIT_DIM dimension table was introduced between the 1st and 2nd stage of the ETL process. Special attention was paid to the manner in which the PADS2_VISIT_DIM was constructed. Firstly, a SELECT DISTINCT statement was used to obtain a unique record set from VISIT_DETAILS. The next step was to populate the VISIT_DIM_KEY in VISIT_DETAILS with the new surrogate key. The reasoning for this sequence of steps can be summarized as follows. The VISIT_DETAILS table contained dimension attributes of VISITS, ACCOUNTS, DISCHARGES, ADMISSIONS and PATIENTS. To construct the VISIT_DIM table from

all these would take longer, than just constructing it from VISIT_DETAILS. After the VISIT_DIM was constructed, conventional SQL DML were used to update the surrogate key VISIT_DIM_KEY in VISIT_DETAILS. The new VISIT_DIM tabled contained 2,567 rows, each row representing a unique combination of the column attributes. See SQL code below.

Inital SQL Code to update the surrogate key

```
UPDATE /*+ NOLOGGING */
VISIT_DETAILS outer
SET VISIT_DIM_KEY =  (SELECT v.VISIT_DIM_KEY   FROM   VISIT_DIM v
  WHERE  v.VISIT_TYPE                        = outer.VISIT_TYPE
  AND    v.DISCHARGE_DISPOSITION             = outer.DISCHARGE_DISPOSITION
  AND    v.VISIT_UPFS_CLASS                  = outer.VISIT_UPFS_CLASS
  AND    v.VISIT_DETAILED_UPFS_CLASS         = outer.VISIT_DETAILED_UPFS_CLASS
  AND    v.VISIT_CLASSIFICATION              = outer.VISIT_CLASSIFICATION
  AND    v.VISIT_PRIVATE_STATE               = outer.VISIT_PRIVATE_STATE
  AND    v.VISIT_MVA                         = outer.VISIT_MVA);
```

An interesting observation was made. The SQL code to update the surrogate key generated very high CPU and I/O costs. It was no surprise when the SQL statement took longer than 15 minutes to complete. A new approach was required and the Oracle BULK collect feature was experimented with. Listed below is the SQL code. RAM limitations on the data warehouse server resulted in segments of 100,000 records each. The total updated set equalled 2,067,508 rows.

Revised SQL Code to update the surrogate key

```
DECLARE  CURSOR c1 IS
   SELECT vd.ADMISSION_KEY, v.VISIT_DIM_KEY
   FROM  VISIT_DIM v, VISIT_DETAILS vd
   WHERE v.VISIT_TYPE                        = vd.VISIT_TYPE
   AND   v.DISCHARGE_DISPOSITION             = vd.DISCHARGE_DISPOSITION
   AND   v.VISIT_UPFS_CLASS                  = vd.VISIT_UPFS_CLASS
   AND   v.VISIT_DETAILED_UPFS_CLASS         = vd.VISIT_DETAILED_UPFS_CLASS
   AND   v.VISIT_CLASSIFICATION              = vd.VISIT_CLASSIFICATION
   AND   v.VISIT_PRIVATE_STATE               = vd.VISIT_PRIVATE_STATE
   AND   v.VISIT_MVA                         = vd.VISIT_MVA
   AND   vd.VISIT_DIM_KEY IS NULL;

  TYPE t_num_array  IS TABLE OF NUMBER(10) INDEX BY BINARY_INTEGER;

  v_admission_key          t_num_array;
  v_visit_dim_key          t_num_array;

BEGIN
  LOOP
    OPEN c1;
    FETCH c1
    BULK COLLECT INTO v_admission_key, v_visit_dim_key LIMIT 100000;
    EXIT WHEN c1%ROWCOUNT = 0;
    CLOSE c1;

    FORALL i IN 1..v_admission_key.count
      UPDATE VISIT_DETAILS
         SET VISIT_DIM_KEY = v_visit_dim_key(i)
       WHERE ADMISSION_KEY = v_admission_key(i);
    COMMIT;
  END LOOP;
END;
```

The Oracle BULK collect statement, using arrays, reduced the update time from the initial 15 minutes to 2 minutes 49 seconds. Listed below are the respective SQL statements, one using VISIT_DIM and the other using VISIT_DETAILS. The SQL syntax changes are highlighted in bold.

Select Statement SQL Code (using VISIT_DIM)

```
SELECT
  t.TREATMENT_LOCATION_NAME,
  vd.VISIT_DETAILED_UPFS_CLASS,
  d.FINANCIAL_YEAR,
  SUM(TOTAL_LEVIES)
FROM VISIT_DIM vd, FLOW_EVENTS e, WH_MATCHING.TREATMENT_LOCATION_DIM t,
WH_MATCHING.DATE_DIM d
WHERE e.VISIT_DIM_KEY = vd.VISIT_DIM_KEY(+)
AND e.FACILITY_KEY = t.TREATMENT_LOCATION_KEY
AND t.TREATMENT_LOCATION_NAME = 'Metsimaholo Hospital (Sasolburg)'
AND e.EVENT_DATE_KEY = d.DATE_KEY
GROUP BY t.TREATMENT_LOCATION_NAME, vd.VISIT_DETAILED_UPFS_CLASS, d.FINANCIAL_YEAR
```

Select Statement SQL Code (using VISIT_DETAILS)

```
SELECT
  t.TREATMENT_LOCATION_NAME,
  v.VISIT_DETAILED_UPFS_CLASS,
  d.FINANCIAL_YEAR,
  SUM(TOTAL_LEVIES)
FROM VISIT_DETAILS v, FLOW_EVENTS e, WH_MATCHING.TREATMENT_LOCATION_DIM t,
WH_MATCHING.DATE_DIM d
WHERE e.ADMISSION_KEY = v.ADMISSION_KEY(+)
AND e.FACILITY_KEY = t.TREATMENT_LOCATION_KEY
AND t.TREATMENT_LOCATION_NAME = 'Metsimaholo Hospital (Sasolburg)'
AND e.EVENT_DATE_KEY = d.DATE_KEY
GROUP BY t.TREATMENT_LOCATION_NAME, v.VISIT_DETAILED_UPFS_CLASS, d.FINANCIAL_YEAR;
```

Hobbs et al. (2005:255) pointed out that the query optimizer is used to determine a plan to execute a query in the fastest possible time. The query optimizer in Oracle Database 10g is known as the cost-based optimizer (Hobbs et al., 2005:256). To verify the different table access paths that the Oracle query optimizer would use for VISIT_DIM and VISIT_DETAILS, a query execution plan was generated with an associated cost. The cost is a measure of how much I/O, CPU time and memory will be required to execute the query using the execution plan (Hobbs et al., 2005:256).

Oracle provides several tools to display the execution plan. When using the **autotrace** option of SQL*Plus, the tool can display the plan when one executes the query. Alternatively, to just get the query plan without executing the query, one can use EXPLAIN PLAN. The output of EXPLAIN PLAN is placed in a table called the PLAN_TABLE in the same database schema. For the purpose of simplicity the EXPLAIN PLAN option was used and depicted below (see figures 6-19 and 6-20) are the execution plans of both SQL statements.

Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time	Pstart	Pstop
0	SELECT STATEMENT		250	17250	474 (20)	00:00:06		
1	HASH GROUP BY		250	17250	474 (20)	00:00:06		
* 2	HASH JOIN		399K	26M	426 (11)	00:00:06		
3	TABLE ACCESS FULL	DATE_DIM	9132	133K	25 (4)	00:00:01		
* 4	HASH JOIN RIGHT OUTER		399K	20M	395 (10)	00:00:05		
5	VIEW	index$_join$_001	2657	29227	7 (15)	00:00:01		
* 6	HASH JOIN							
7	BITMAP CONVERSION TO ROWIDS		2657	29227	1 (0)	00:00:01		
8	BITMAP INDEX FULL SCAN	VISIT_DIM_BMX4						
9	INDEX FAST FULL SCAN	VISIT_DIM_PK	2657	29227	5 (0)	00:00:01		
10	NESTED LOOPS		399K	16M	382 (8)	00:00:05		
* 11	TABLE ACCESS FULL	TREATMENT_LOCATION_DIM	1	26	6 (0)	00:00:01		
12	PARTITION LIST ITERATOR		399K	6636K	376 (8)	00:00:05	KEY	KEY
* 13	TABLE ACCESS FULL	FLOW_EVENTS	399K	6636K	376 (8)	00:00:05	KEY	KEY

Figure 6-19: SQL Execution Plan Using VISIT_DIM

Id	Operation	Name	Rows	Bytes	TempSpc	Cost (%CPU)	Time	Pstart	Pstop
0	SELECT STATEMENT		175	12250		12118 (3)	00:02:26		
1	HASH GROUP BY		175	12250		12118 (3)	00:02:26		
* 2	HASH JOIN		399K	26M		12069 (3)	00:02:25		
3	TABLE ACCESS FULL	DATE_DIM	9132	133K		25 (4)	00:00:01		
* 4	HASH JOIN OUTER		399K	20M	21M	12038 (3)	00:02:25		
5	NESTED LOOPS		399K	17M		382 (8)	00:00:05		
* 6	TABLE ACCESS FULL	TREATMENT_LOCATION_DIM	1	26		6 (0)	00:00:01		
7	PARTITION LIST ITERATOR		399K	7417K		376 (8)	00:00:05	KEY	KEY
* 8	TABLE ACCESS FULL	FLOW_EVENTS	399K	7417K		376 (8)	00:00:05	KEY	KEY
9	VIEW	index$_join$_001	2075K	19M		8348 (3)	00:01:41		
* 10	HASH JOIN								
11	INDEX FAST FULL SCAN	VISIT_DET_ADMISSION_KEY	2075K	19M		2860 (3)	00:00:35		
12	BITMAP CONVERSION TO ROWIDS		2075K	19M		79 (0)	00:00:01		
13	BITMAP INDEX FULL SCAN	VISIT_DET_UPFSCLASS_BMX							

Figure 6-20: SQL Execution Plan using VISIT_DETAILS

By examining the execution plans illustrated in the above figures it is clear that the time needed is **6 seconds** for the VISIT_DIM query compared to **2 minutes 26 seconds** for the VISIT_DETAILS query

Another important observation is the Cost (% CPU) of **474** for the VISIT_DIM query compared to **12,118** for the VISIT_DETAILS query. From these two important observations one must conclude it is imperative to evaluate SQL execution plans when construction dimensional models.

Construction of FLOW_EVENTS

By introducing a partitioned fact table one can improve performance of queries against that table. Hobbs et al. (2005:114) argues that an expensive full table scan can be avoided by using partitions and this feature is generally known as **Partition Elimination** or **Dynamic Partition Pruning in the database world.**

The fact table FLOW_EVENTS was constructed as a LIST partitioned table using FACILITY_KEY as the partition key. According to Hobbs et al. (2005:119) if a table is list partitioned, the optimizer can perform partition pruning if the query asks for a range or list of partition-key values. Why use a partitioned table as fact table? Simple because the users will interface with the ad hoc query tool by querying their institutional data only. Taking this user query pattern into account together with dynamic partition pruning, the query will only be reading from a certain partition instead of a full table scan on the fact table containing nearly 11 million rows. See Appendix D for the SQL DDL to create the FLOW_EVENTS partitioned table.

6.8.3. Dimensional Model

A dimensional model (see figure 6-21) was constructed by arranging all the relevant dimension and fact tables into a star schema to be used by the Cognos Query tool. Tables were renamed to increase ease-of-use and improved user navigation in the dimensional model. See table 6-5 for name changes in the metadata for the dimensional model.

Table 6-5: Data warehouse and Dimensional model table names

Data Warehouse Table Name	Dimensional Model Table Name
FLOW_EVENTS	PADS2_EVENTS_FACT
VISIT_DIM	PADS2_VISIT_DIM
PATIENTS	PADS2_PATIENTS_DIM
ACCOUNTHOLDERINFO	PADS2_ACCOUNTHOLDER_DIM
substract of (VISIT_DETAILS)	PADS2_ACCOUNTS_DIM
substract of (DATE_DIM)	PADS2_EVENTDATE_DIM
substract of (TREATMENT_LOCATION_DIM)	PADS2_TREATMENT_LOCATION_DIM

The *role-playing dimension* PADS2_TREATMENT_LOCATION_DIM was added to the dimensional model as a database view of the TREATMENT_LOCATION_DIM dimension table (using the filter PADS_FLAG='TRUE'). The *role-playing dimension* PADS2_EVENTDATE_DIM was implemented as a database view from the DATE_DIM dimension table and introduced as the time dimension.

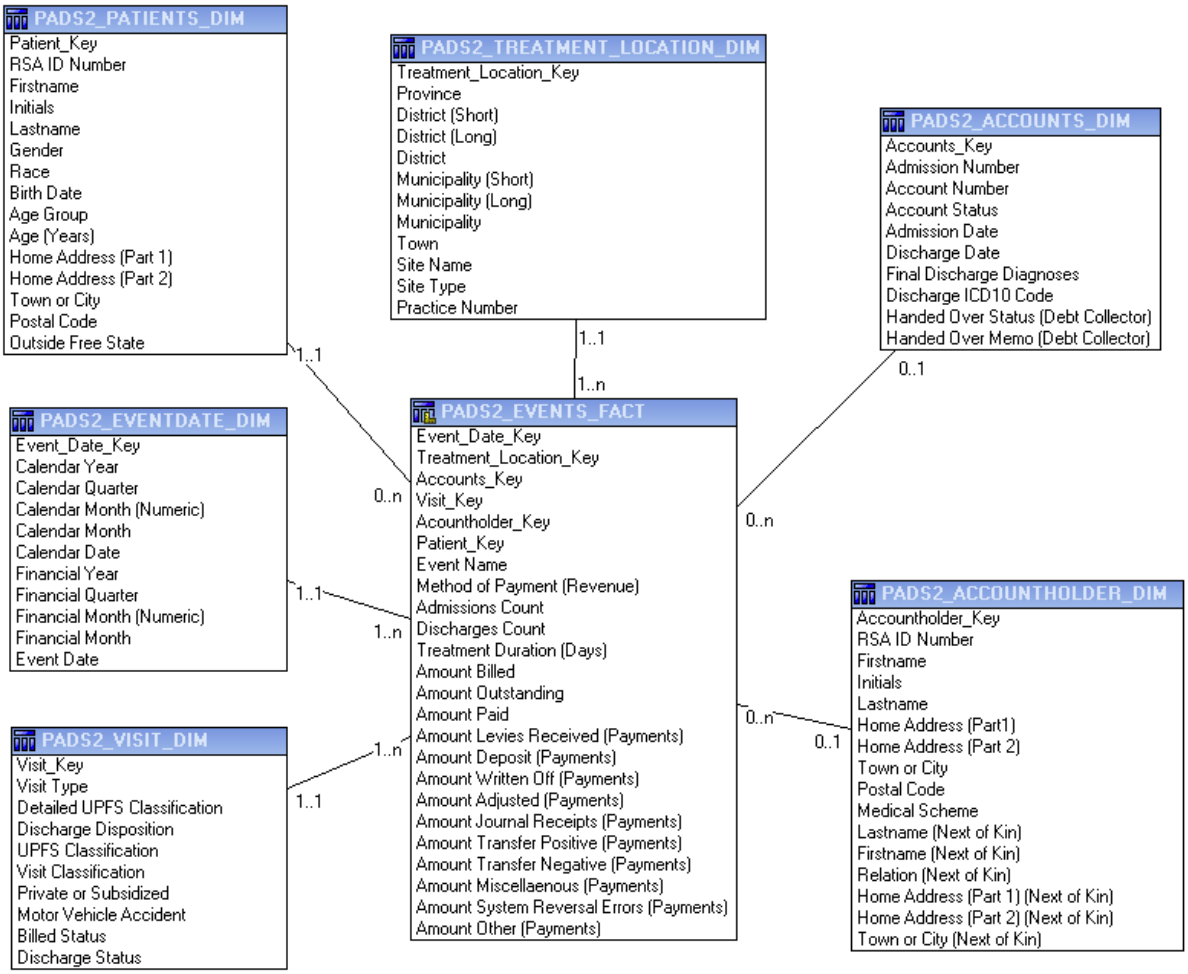


Figure 6-21: PADSDM Dimensional Model

6.9. Development of the TB Data Mart (TBDM)

TB treatment is an important aspect to monitor for patients on ARV treatment. One needs to be reminded that ARV patients are not clinically allowed to be on both TB and ARV treatments simultaneously. When a clinician or nurse detects that a patient on ARV treatment has contracted TB, the patient will be referred immediately for TB treatment. Thereafter the patient's ARV treatment will be suspended until the TB has been cured. This could take up to six (6) months. The patient can only recommence ARV therapy, once the TB treatment centre has confirmed that the TB has been cured.

This data mart would contain the clinical and strategic information of patients on TB treatment. Identified fields within the ARVDM will be updated with selected information extracted from the TBDM to build up the clinical picture of a patient.

6.9.1. Addressing Managerial Outcomes

The majority of indicators and reporting needs are destined for the National Department of Health and the World Health Organization (WHO).

Internal Management Requirements:

- Identify patients that are part of the TB programme and also enrolled in the ARV programme

External Management Requirements:

- Patients with new smear positive cure rate target 75%
- Patients with new smear successful treatment target 85%
- Patients with new smear positive smear conversion at 2 months 75%
- Patients with sputum smear turnaround time (TAT)
    - Internal 80% within 72 hours
    - National 80% within 46 hours
- DOT (supervision) 90% of all TB patients

6.9.2. Extraction, Transformation and Loading Challenges

The operational data source of the TBDM consisted of a single de-normalized table (1NF) that was made available externally as a CVS (comma delimited) file. In order to construct a dimensional model from this single table, the TB_PATIENTS_DIM table was constructed from unique patient entries.

Thereafter a *junk dimension* called TB_JUNK_DIM was implemented with all possible combinations of Smear conversions, Regimens, Categories, Classifications and Treatment Outcomes. The reason for a junk dimension was the fact that these indicators don't logically belong to the core dimension tables. The approach followed to create the junk dimension, was to create the table in advance, with each possible unique combination generating a row. The reason for this approach was that the data mart used a REFRESH ETL approach, meaning all the dimension and fact tables are cleaned and then populated. A surrogate key JUNK_KEY was generated and used in the fact table TB_FACT. Ross (2003) also points out that if the number of rows in a junk dimension approaches or exceeds the number of rows in the fact table, the design should be re-evaluated. For the TBDM this step is not needed as the rows in the fact table (r=128 086) still exceeds the rows in the junk table (r=137).

The *role-playing dimension* TB_TREATMENT_LOCATION_DIM was added to the dimensional model as a database view of the TREATMENT_LOCATION_DIM dimension table (using the filter TB_FLAG='TRUE'). The *role-playing dimensions* TB_REGISTRATION_DATE_DIM and TB_TREATMENT_DATE_DIM were implemented as database views from the DATE_DIM dimension table and introduced as time dimensions. See depicted below (figure 6-22) the final dimensional model for the TB data mart.

Next the TB_PATIENTS_DIM dimension was added to the dimensional model. Due to the lack of identifiers in the operational data source (and exported CVS file), each fact had to be linked to its corresponding patient entry. In practice this meant that the number of records in the fact table and the dimension table was exactly the same (r=128 086). TB attributes was added to each patient, in an effort to try and simplify identifying a patient, especially where the First name, Surname and Age were exactly the same. In future chapters the challenge to standardizing a patient dimension will be discussed in an effort to create a more conformed patient dimension.

Finally, with the dimension and junk tables in place, the fact table TB_FACT was created. To avoid the situation of a *factless* fact table, a surrogate fact column called ***treatment_count*** was inserted into the fact table. The value of the column was 1 for each row.

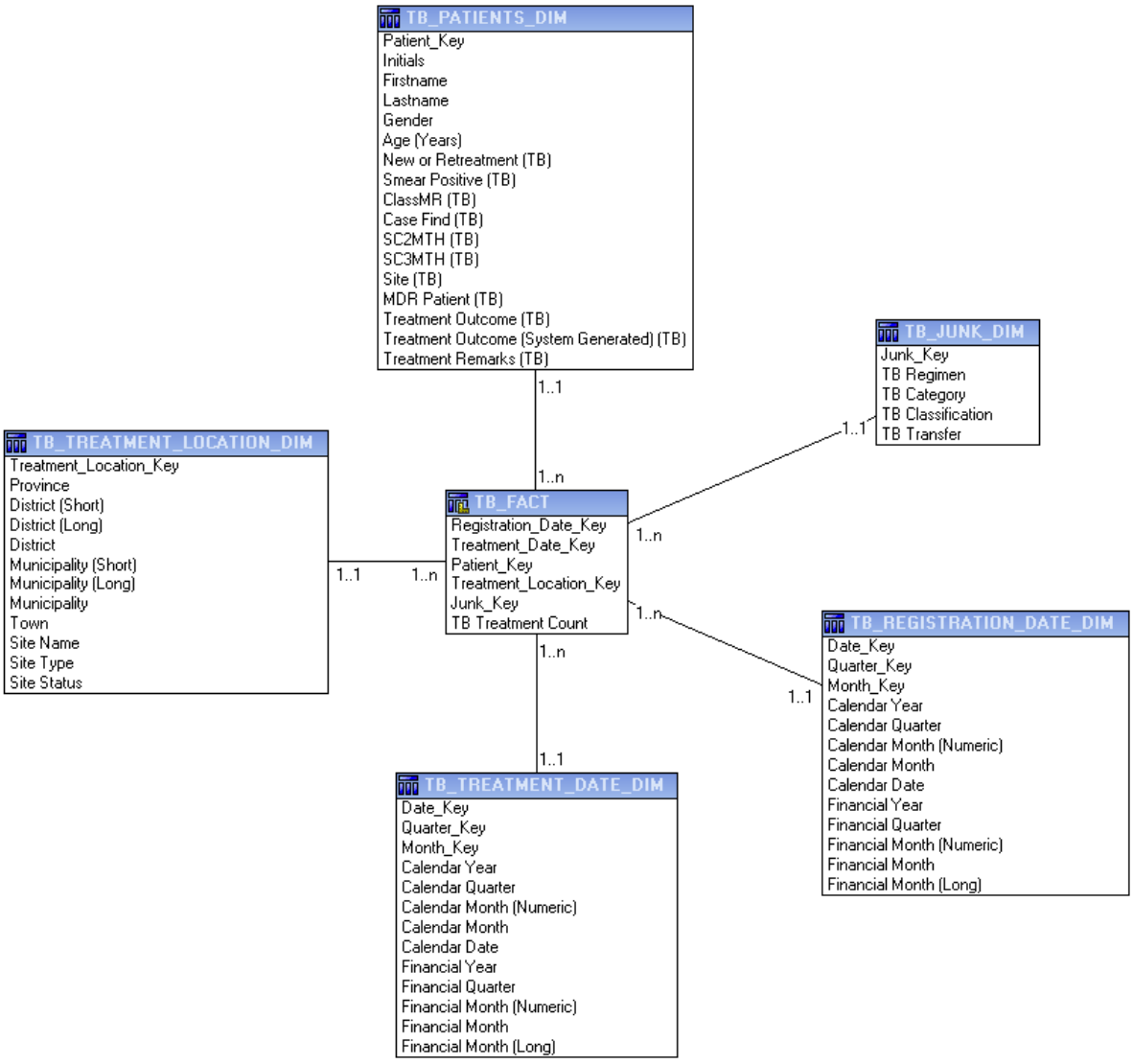


Figure 6-22: TBDM Dimensional Model

6.10. Development of the Notifiable Diseases Data Mart (NDDM)

This data mart's primary goal was to provide strategic information from the notifiable diseases (ND) transactional system in terms of *tuberculosis* and *diarrhea* which are both closely associated with HIV and AIDS. The operational data source contained a single de-normalized table (1NF) for all patient demographics as well as their diseases. Data has been collected since March 2003 of all patients with notifiable diseases in the Free State and was a valuable data source to add to the FSDOH data warehouse.

6.10.1. Addressing Managerial Outcomes

Internal Management Requirements:

Only three indicators are required as an internal management requirement. The indicators are as follows:

- Case fatality rate for *cholera* = deaths due to cholera / reported cholera cases x 100
- Case fatality date for *diarrhea* = deaths due to diarrhea / reported diarrhea cases x 100
- Case fatality rate for *malaria* = deaths due to malaria/ reported malaria cases x 100

External Management Requirements: None.

6.10.2. Extraction, Transformation and Loading Challenges

A design challenge presented itself when the operational data source's table definitions were studied. The operational data source did not adhere to the traditional database design approach of using table normalization (3NF). Instead, the main table in the operational data source was designed using table attributes as ***rows***. Traditional table normalization techniques suggest using table columns to represent attributes. See figure 6-23 for the operational data source database design layout.

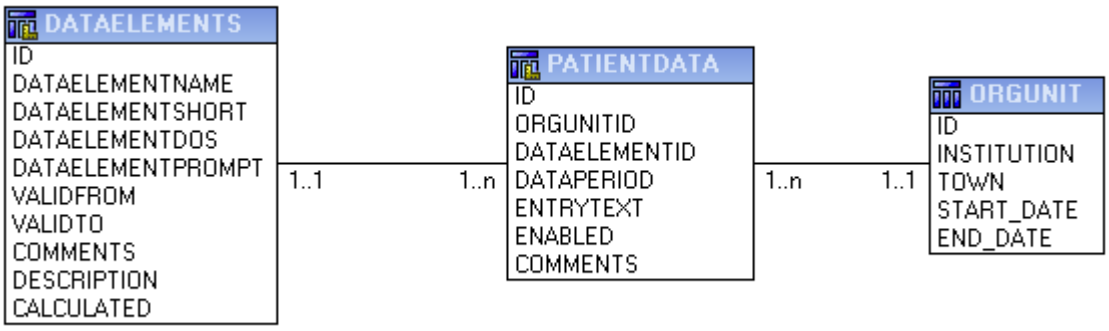


Figure 6-23: Database Schema of Notifiable Diseases

The first challenge was to convert the de-normalized PATIENTDATA table, to a table using columns for attributes instead of rows. Staging tables called WH_PasientData_Pivot and WH_NotifData_Pivot were created in the staging area to assist with this conversion process. These staging tables provided the mechanism to split the demographic information and the disease information also into two different tables. See figure 6-24 for the staging area database design layout.

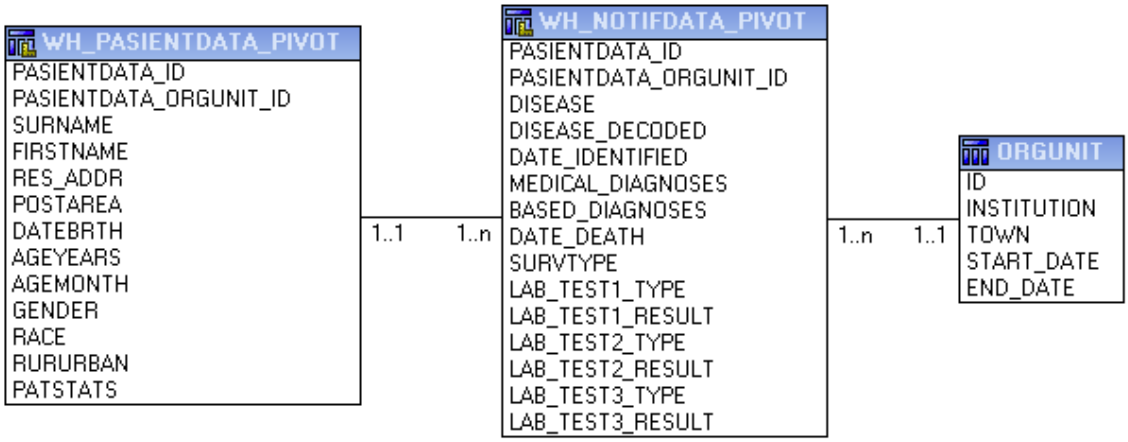


Figure 6-24: Database Staging Tables for Notifiable Diseases

In order to construct these above depicted staging tables, Oracle 10g's analytical functions such as 'ROW_NUMBER() OVER' and 'PARTITION BY' were used. See Appendix E (part 1 and part 2) for the SQL code to construct a column attribute table from a row attribute table.

The next step was to load and convert these staging tables (WH_NotifData_Pivot and WH_PatientData_Pivot) into a dimensional model. The NOTIF_FACT fact table was extracted from WH_NotifData_Pivot staging table while the NOTIF_PATIENTS_DIM was extracted from the WH_PatientData_Pivot staging table. Lookup dimension tables NOTIF_DISEASE_DIM and NOTIF_PATIENTS_TEST_DIM were created to improve the navigation ability of the dimensional model.

A *junk dimension* called NOTIF_DIAGNOSES_JUNK_DIM was introduced to capture attributes that did not fit into any other dimension table.

Finally, the *role-playing dimension* NOTIF_TREATMENT_LOCATION_DIM was added to the dimensional model as a database view of the TREATMENT_LOCATION_DIM dimension table (using the filter NOTIF_FLAG='TRUE'). The *role-playing dimensions* NOTIF_DEATH_DATE_DIM and NOTIF_IDENTIFIED_DATE were implemented as database views from the DATE_DIM dimension table and introduced as time dimensions. See depicted below (figure 6-25) the final dimensional model for the notifiable diseases data mart.

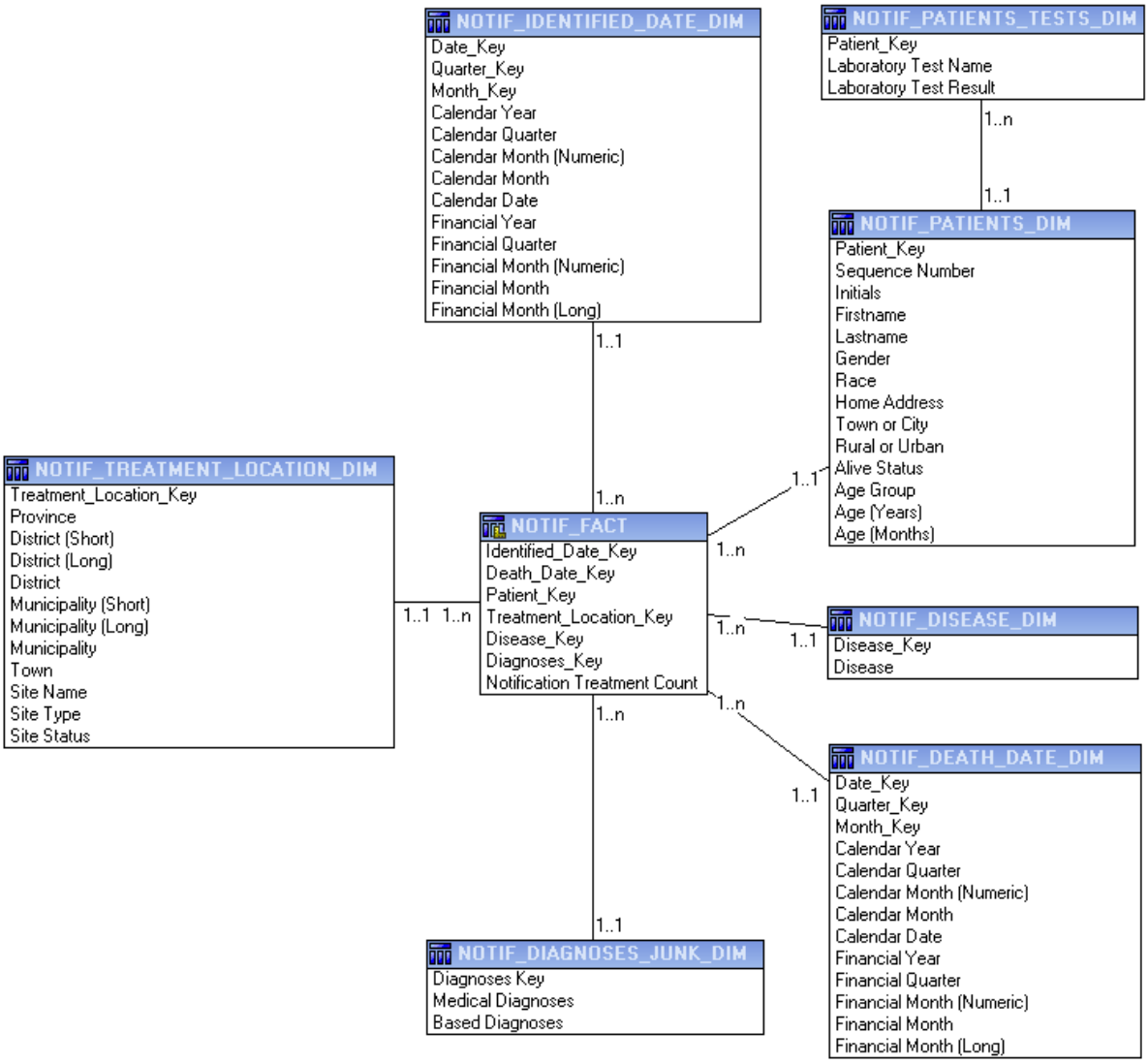


Figure 6-25: NOTIFDM Dimensional Model

6.11. Chapter Summary

This chapter examined and documented the construction of the FSDOH data warehouse with all its individual data marts. The following chapter will provide an overview of the decisions and implementation methodology behind the FSDOH BI solution.

CHAPTER 7 - INTEGRATED BUSINESS INTELLIGENCE SOLUTION

7.1. Introduction

The previous chapter covered the process of building a single **appropriately designed** and **integrated** data warehouse that was required to provide a cost-effective business analysis tool for decision-making.

This chapter will provide an overview of the decisions that created the environment for transforming the data warehouse into a business intelligence (BI) solution. This chapter forms part of the *action taking* phase of the action research cycle

7.2. Turning the Data Warehouse into Business Intelligence

According to Mallach (2000:524), the contents of data warehouse can be analyzed in two distinct classifications: active analysis and automated analyses. By viewing these classifications from the business intelligence point of view, Berndt et al. (2000) refer to active analysis as “navigation activities” and automated analyses as “summarization activities” of business intelligence (BI).

Using user queries and OLAP can be seen as methods of active analysis. Mallach (2000:531) states that many good decisions can be on the basis of a series of questions (**queries**) and answers. Data warehouse analysis software allows this “slicing and dicing” of content in answering questions. If a user needs deeper detail, he or she will drill-down until the necessary decisions have been made. This mode of operation is often referred to as OLAP. In automated analyses, *the computer does the work*, as opposed to the active user-guided analysis. The user need only instruct the computer what he or she wants to find out in the analysis. Mallach (2000:524) notes that data mining is the most popular form of automated analyses. Decision trees, neural networks, nearest neighbor approaches are all examples of data mining methods.

The BI world is, however, changing fast because data analysis is getting more complex. Whitehorn (2007) made the argument that BI should now be characterized in three generations to enable users to identify the trends and select the appropriate technology:

- The first generation was about understanding the past.
- The second generation was about analyzing why things happened and making recommendations about the future
- The third generation is about making information available to the people in front of the customer.

From these characterizations, Whitehorn (2007) concluded that first- and second-generation BI systems needed to support a limited number of people who ran large, complex analytical queries. The third generation must support not only more complex queries from the same analysts, but also a new workload consisting of thousands of users running very different queries. These may well be complex, but each is likely to hit a relatively small set of data within the warehouse (Whitehorn, 2007).

7.3. FSDOH BI Solution

The FSDOH opted to concentrate on ***navigation activities*** or ***active analysis*** as the primary method of providing the necessary strategic information. By taking the trends identified by Whitehorn (2007) into account, it was clear that understanding the past would assist the management team in addressing the challenges the ART programme presented This meant that the FSDOH needed to investigate a first-generation BI tool.

The next point was the BI tool's functionality itself. Lawyer and Chowdhury (2004) argued in their work that when using the Kimball approach in constructing a data warehouse, authoring SQL to access data arranged in a multidimensional database would be a very complex task. In order to handle the capabilities of drilling up, down, and sideways within a multidimensional structure, the business user typically requires a BI tool that includes OLAP functionality. Berndt and Fisher (2001) expanded on this view by stating that OLAP tools take advantage of the predictable dimensional structure of data warehouse components, using the dimensions to constrain or group the query results and summarizing selected numeric facts in a star join query. Clearly from this we can conclude that OLAP is a necessary analytical tool in accessing the FSDOH data warehouse.

The next outstanding issue was the choice of OLAP platform to use. Gorla (2003) suggested in his work that multidimensional OLAP (MOLAP) be used for non-sophisticated computer users and relational OLAP (ROLAP) for sophisticated users. The MOLAP model is best implemented by storing the data multidimensionally, that is, easily viewable in a multidimensional way (Ponniah, 2001:365) using the format of a cube (Verma and Harper, 2001). March and Hevner (2007) emphasizes the point that OLAP cubes play an important role in providing managers with an analysis tool for specific decision making situations. By making the assumption that the FSDOH users can be classified as **non-sophisticated** users, the MOLAP architecture with cubes would be best suited for the FSDOH.

The final point was the choice of human-computer interface. Ewen et al. (1998) found during their research study that a server-side BI tool implementation with **Web access** is less expensive than buying multiple client licenses and leverages server resources for faster performance. This viewpoint was shared by De Beer (2006) who viewed Web-Based OLAP (WOLAP) as the next generation BI tool providing "thin-client" viewing tools for analyzing information. The ideal OLAP environment for the FSDOH will therefore be a mixture of the MOLAP and WOLAP architectures.

The Cognos BI Suite fitted all these above mentioned requirements. The *Cognos PowerPlay Enterprise Server* (PPES) uses WOLAP as front-end while the underlying back-end architecture is based on MOLAP. *Cognos ReportNet* offered an ad-hoc reporting tool to display grids and graphs and to provide drill-through capabilities. The *Cognos 7.3 BI Suite* was implemented in April 2005 and included *Cognos Transformer* to construct OLAP cubes, *Cognos PPES Server* and *Cognos ReportNet.*

In August 2007 the Cognos BI platform was upgraded from version 7.3 to version 8. This upgrade provided the additional functionality of Metrics Manager and Event Studio. Metrics Manager is the main tool for developing any Scorecard or Dashboard applications. Event studio on the other hand is a *"push"* technology, and differs from the traditional *"pull"* technology associated with business intelligence and can trigger events based on metric (or measure) thresholds in the data warehouse or OLAP cubes.

7.4. Constructing the FSDOH BI Solution

The FSDOH BI Solution can be divided into two distinct sections namely: ad-hoc query and OLAP or cube analysis. *Ad-Hoc query* functionality was made available using Cognos Query Studio and Cognos Report Studio. *OLAP or cube analysis* functionality was made available using the WOLAP front-end interface of the Cognos PowerPlay Enterprise Server.

Before any Cognos Report and Cube can be constructed, models had to be created based on the metadata in the underlying dimensional model contained within the data warehouse. A model serves as an insulating layer between Cognos 8 reporting, users and the database. Packages are model subsets that ensure users are provided with data appropriate for the reporting they need to do, and that the data is structured in ways that make sense from a business perspective (Cognos, 2006). For Cognos 8 reporting, models and packages are created using Cognos Framework Manager.

Summarized below are the steps to follow when creating a model (Cognos, 2006):

- specify the metadata to import from data sources defined in the Cognos 8 content store
- refine the metadata by adding business names, descriptions, multilingual values, calculations, filters, and other components
- specify joins and relationships
- organize the data for presentation in ways that make sense to business users and add value to your databases
- add security to the metadata to determine who can use the metadata and for what purposes

After a model is created, packages are published to locations on the Cognos 8 server where report authors can use them to create reports. A different package, containing only the necessary

information, can be published for each group of users (Cognos, 2006). Depicted in figure 7-1 is the complete reporting and analysis workflow for Cognos 8.

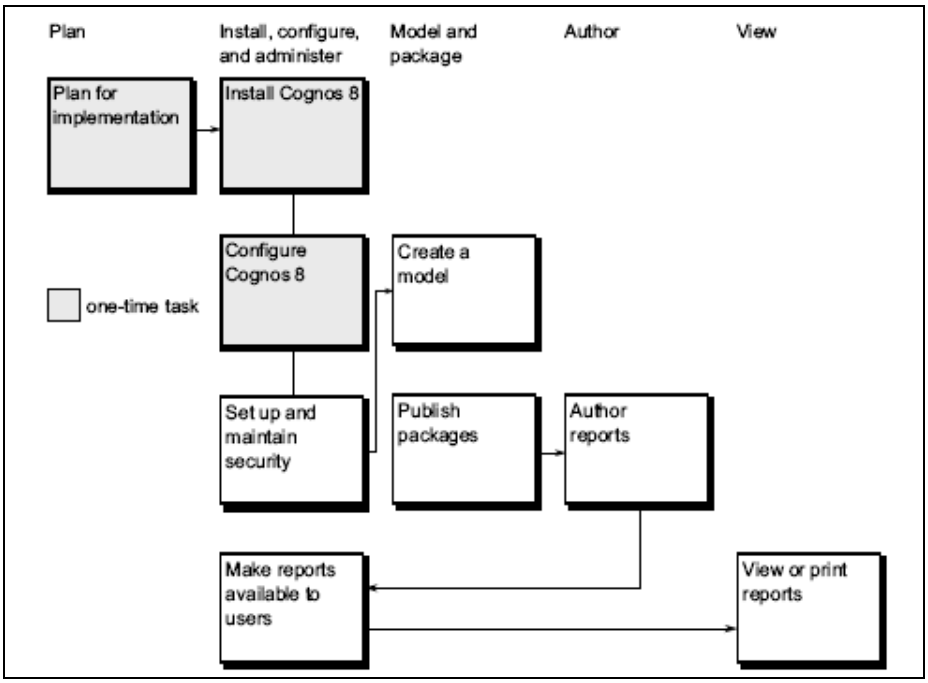


Figure 7-1: Complete Reporting and Analysis workflow for Cognos 8

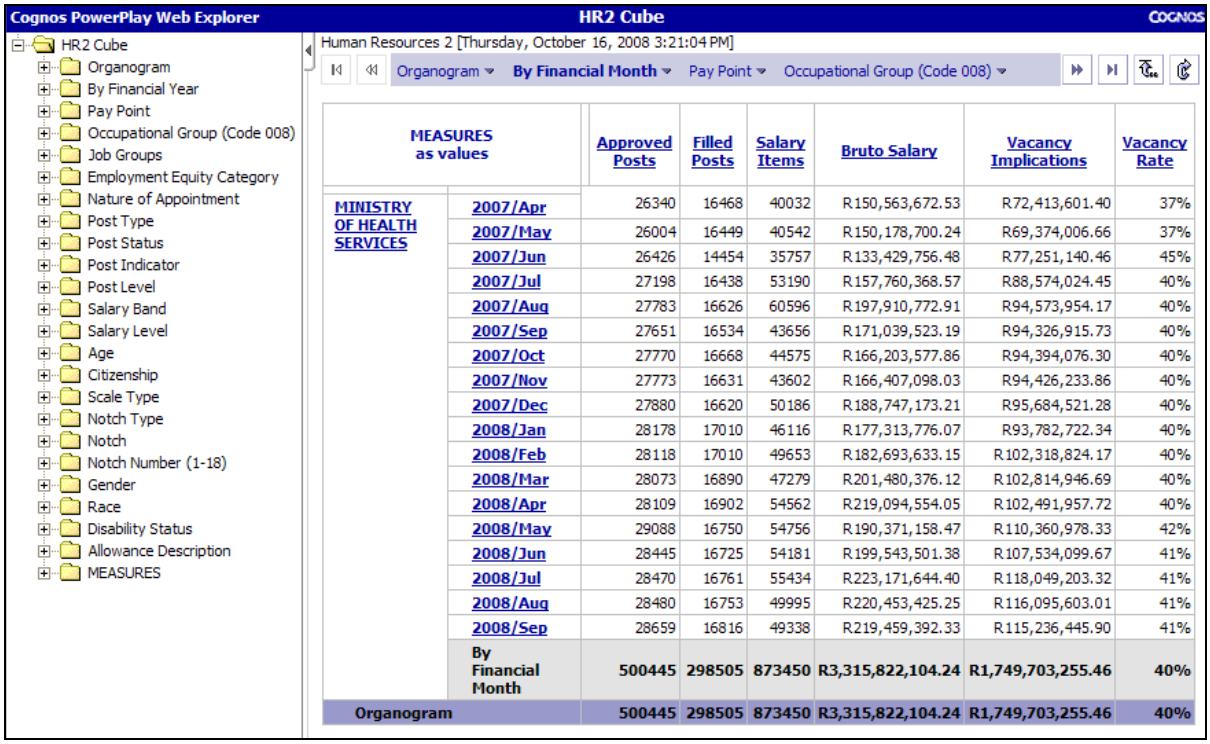
By using the above described workflow process, the author was able to develop the following Cognos metadata, report and cubes that will be discussed in the following sections.

7.4.1. Human Resource Data Mart (HRDM)

Listed below are the Cognos cubes that were constructed from the HRDM:

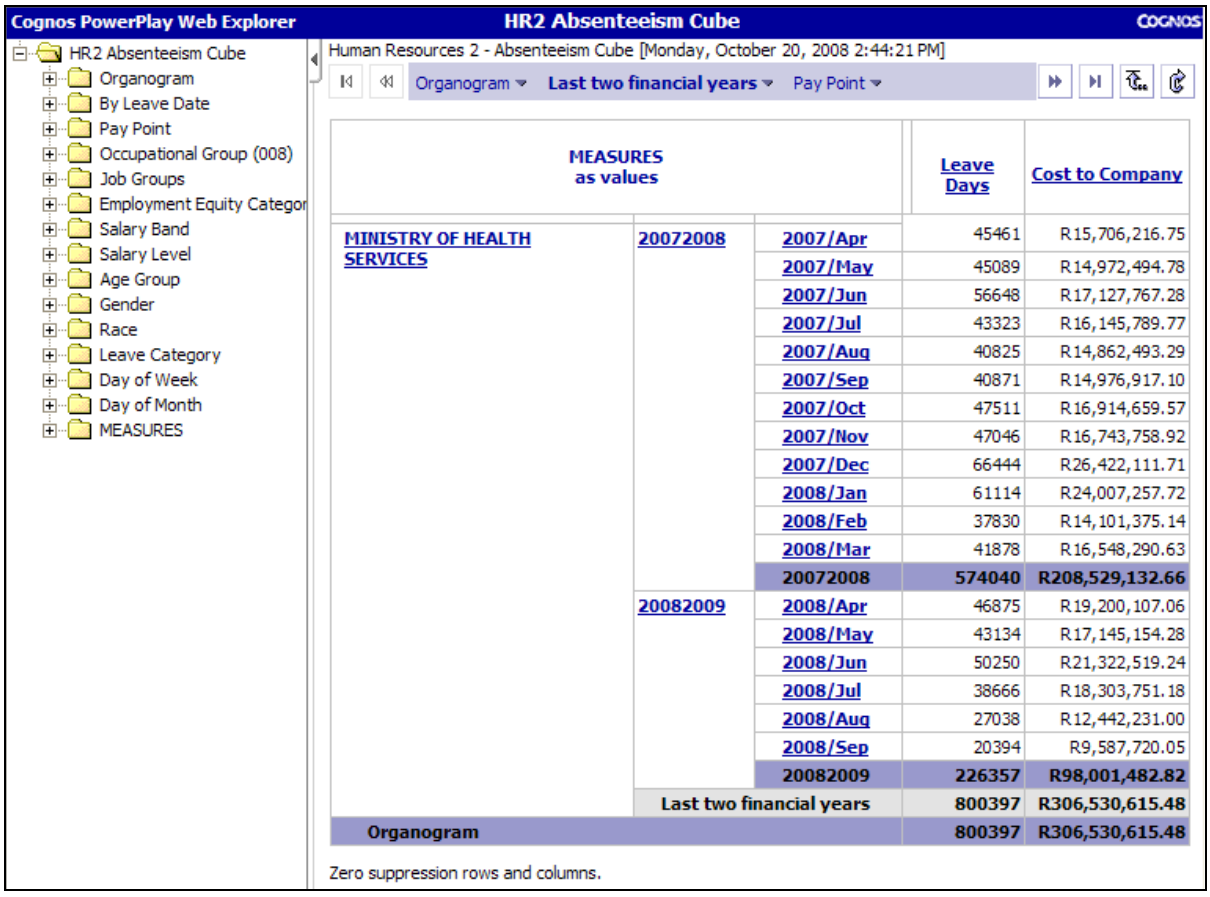
- Staff Establishment Cube
- Absenteeism Cube
- Leave Credits Cube
- Qualifications Cube

Each individual cube focused on a particular subject area within HR in an attempt to simplify user navigation. The Staff Establishment cube had a drill through functionality built-in to list the names and post details of a particular cell within the cube. Figures 7-2 to 7-5 provide screenshots of the listed cube's measures and dimensions as available in Cognos PowerPlay Web Explorer.



MEASURES as values		Approved Posts	Filled Posts	Salary Items	Bruto Salary	Vacancy Implications	Vacancy Rate
MINISTRY OF HEALTH SERVICES	2007/Apr	26340	16468	40032	R150,563,672.53	R72,413,601.40	37%
	2007/May	26004	16449	40542	R150,178,700.24	R69,374,006.66	37%
	2007/Jun	26426	14454	35757	R133,429,756.48	R77,251,140.46	45%
	2007/Jul	27198	16438	53190	R157,760,368.57	R88,574,024.45	40%
	2007/Aug	27783	16626	60596	R197,910,772.91	R94,573,954.17	40%
	2007/Sep	27651	16534	43656	R171,039,523.19	R94,326,915.73	40%
	2007/Oct	27770	16668	44575	R166,203,577.86	R94,394,076.30	40%
	2007/Nov	27773	16631	43602	R166,407,098.03	R94,426,233.86	40%
	2007/Dec	27880	16620	50186	R188,747,173.21	R95,684,521.28	40%
	2008/Jan	28178	17010	46116	R177,313,776.07	R93,782,722.34	40%
	2008/Feb	28118	17010	49653	R182,693,633.15	R102,318,824.17	40%
	2008/Mar	28073	16890	47279	R201,480,376.12	R102,814,946.69	40%
	2008/Apr	28109	16902	54562	R219,094,554.05	R102,491,957.72	40%
	2008/May	29088	16750	54756	R190,371,158.47	R110,360,978.33	42%
	2008/Jun	28445	16725	54181	R199,543,501.38	R107,534,099.67	41%
	2008/Jul	28470	16761	55434	R223,171,644.40	R118,049,203.32	41%
	2008/Aug	28480	16753	49995	R220,453,425.25	R116,095,603.01	41%
	2008/Sep	28659	16816	49338	R219,459,392.33	R115,236,445.90	41%
	By Financial Month	500445	298505	873450	R3,315,822,104.24	R1,749,703,255.46	40%
Organogram		500445	298505	873450	R3,315,822,104.24	R1,749,703,255.46	40%

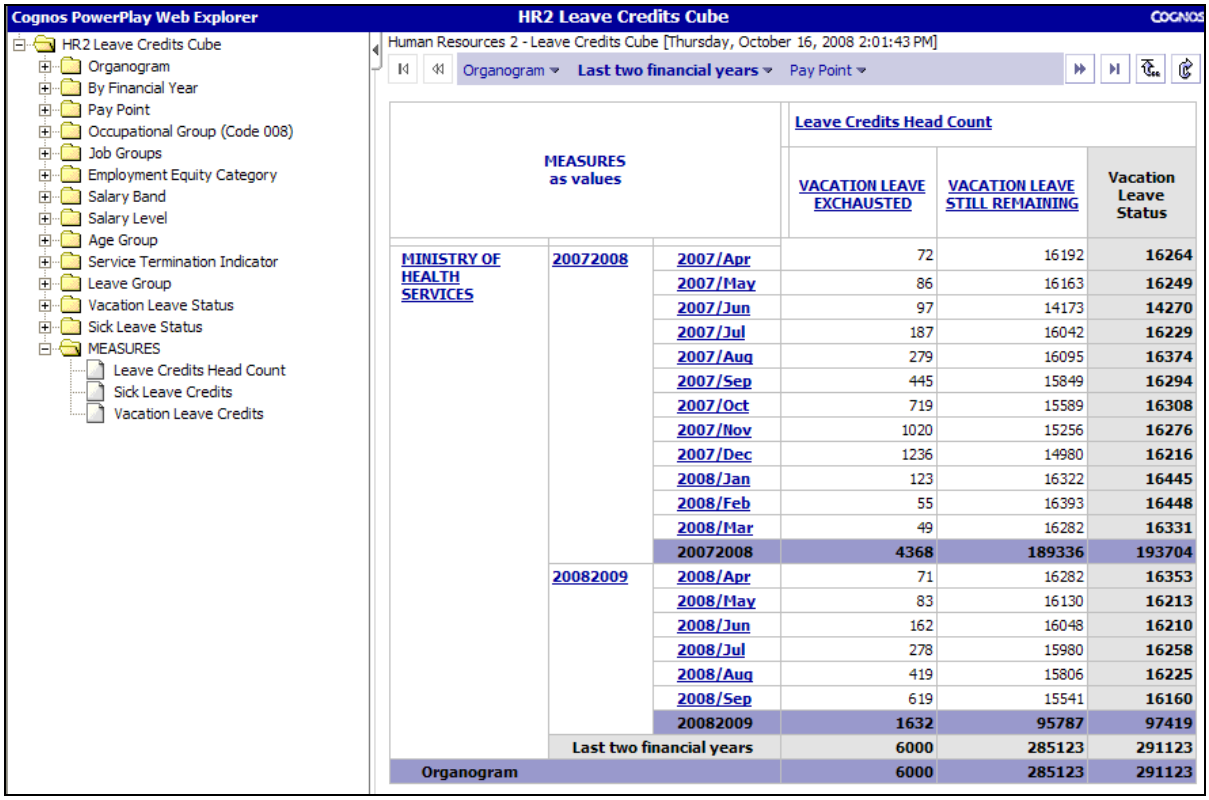
Figure 7-2: Staff Establishment Cube



MEASURES as values			Leave Days	Cost to Company
MINISTRY OF HEALTH SERVICES	20072008	2007/Apr	45461	R15,706,216.75
		2007/May	45089	R14,972,494.78
		2007/Jun	56648	R17,127,767.28
		2007/Jul	43323	R16,145,789.77
		2007/Aug	40825	R14,862,493.29
		2007/Sep	40871	R14,976,917.10
		2007/Oct	47511	R16,914,659.57
		2007/Nov	47046	R16,743,758.92
		2007/Dec	66444	R26,422,111.71
		2008/Jan	61114	R24,007,257.72
		2008/Feb	37830	R14,101,375.14
		2008/Mar	41878	R16,548,290.63
		20072008	574040	R208,529,132.66
	20082009	2008/Apr	46875	R19,200,107.06
		2008/May	43134	R17,145,154.28
		2008/Jun	50250	R21,322,519.24
		2008/Jul	38666	R18,303,751.18
		2008/Aug	27038	R12,442,231.00
		2008/Sep	20394	R9,587,720.05
		20082009	226357	R98,001,482.82
	Last two financial years		800397	R306,530,615.48
Organogram			800397	R306,530,615.48

Zero suppression rows and columns.

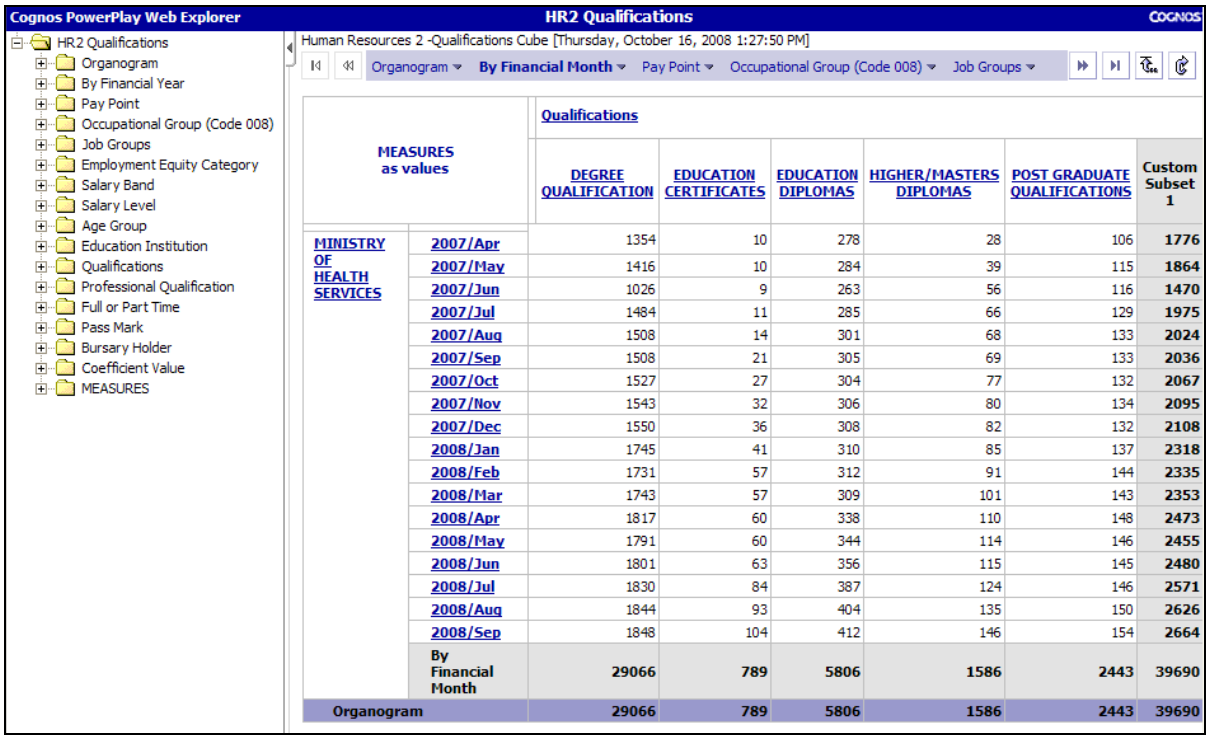
Figure 7-3: Absenteeism Cube



MEASURES as values

			Leave Credits Head Count		
			VACATION LEAVE EXCHAUSTED	VACATION LEAVE STILL REMAINING	Vacation Leave Status
MINISTRY OF HEALTH SERVICES	20072008	2007/Apr	72	16192	16264
		2007/May	86	16163	16249
		2007/Jun	97	14173	14270
		2007/Jul	187	16042	16229
		2007/Aug	279	16095	16374
		2007/Sep	445	15849	16294
		2007/Oct	719	15589	16308
		2007/Nov	1020	15256	16276
		2007/Dec	1236	14980	16216
		2008/Jan	123	16322	16445
		2008/Feb	55	16393	16448
		2008/Mar	49	16282	16331
		20072008	4368	189336	193704
	20082009	2008/Apr	71	16282	16353
		2008/May	83	16130	16213
		2008/Jun	162	16048	16210
		2008/Jul	278	15980	16258
		2008/Aug	419	15806	16225
		2008/Sep	619	15541	16160
		20082009	1632	95787	97419
	Last two financial years		6000	285123	291123
Organogram			6000	285123	291123

Figure 7-4: Leave Credits Cube



MEASURES as values

		Qualifications					
		DEGREE QUALIFICATION	EDUCATION CERTIFICATES	EDUCATION DIPLOMAS	HIGHER/MASTERS DIPLOMAS	POST GRADUATE QUALIFICATIONS	Custom Subset 1
MINISTRY OF HEALTH SERVICES	2007/Apr	1354	10	278	28	106	1776
	2007/May	1416	10	284	39	115	1864
	2007/Jun	1026	9	263	56	116	1470
	2007/Jul	1484	11	285	66	129	1975
	2007/Aug	1508	14	301	68	133	2024
	2007/Sep	1508	21	305	69	133	2036
	2007/Oct	1527	27	304	77	132	2067
	2007/Nov	1543	32	306	80	134	2095
	2007/Dec	1550	36	308	82	132	2108
	2008/Jan	1745	41	310	85	137	2318
	2008/Feb	1731	57	312	91	144	2335
	2008/Mar	1743	57	309	101	143	2353
	2008/Apr	1817	60	338	110	148	2473
	2008/May	1791	60	344	114	146	2455
	2008/Jun	1801	63	356	115	145	2480
	2008/Jul	1830	84	387	124	146	2571
	2008/Aug	1844	93	404	135	150	2626
	2008/Sep	1848	104	412	146	154	2664
	By Financial Month	29066	789	5806	1586	2443	39690
Organogram		29066	789	5806	1586	2443	39690

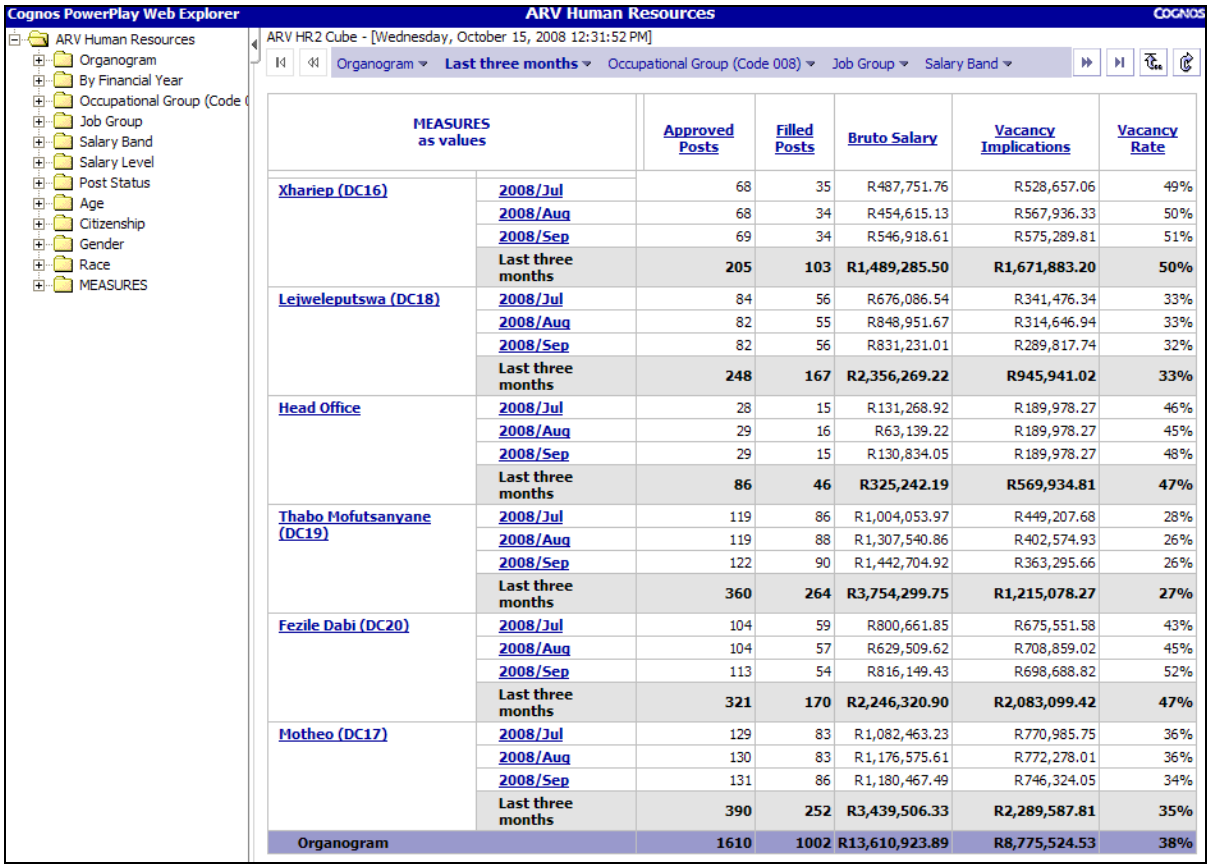
Figure 7-5: Qualifications Cube

7.4.2. Antiretroviral Human Resource Data Mart (ARVHRDM)

Listed below are the Cognos cubes that were constructed from the ARVHRDM:

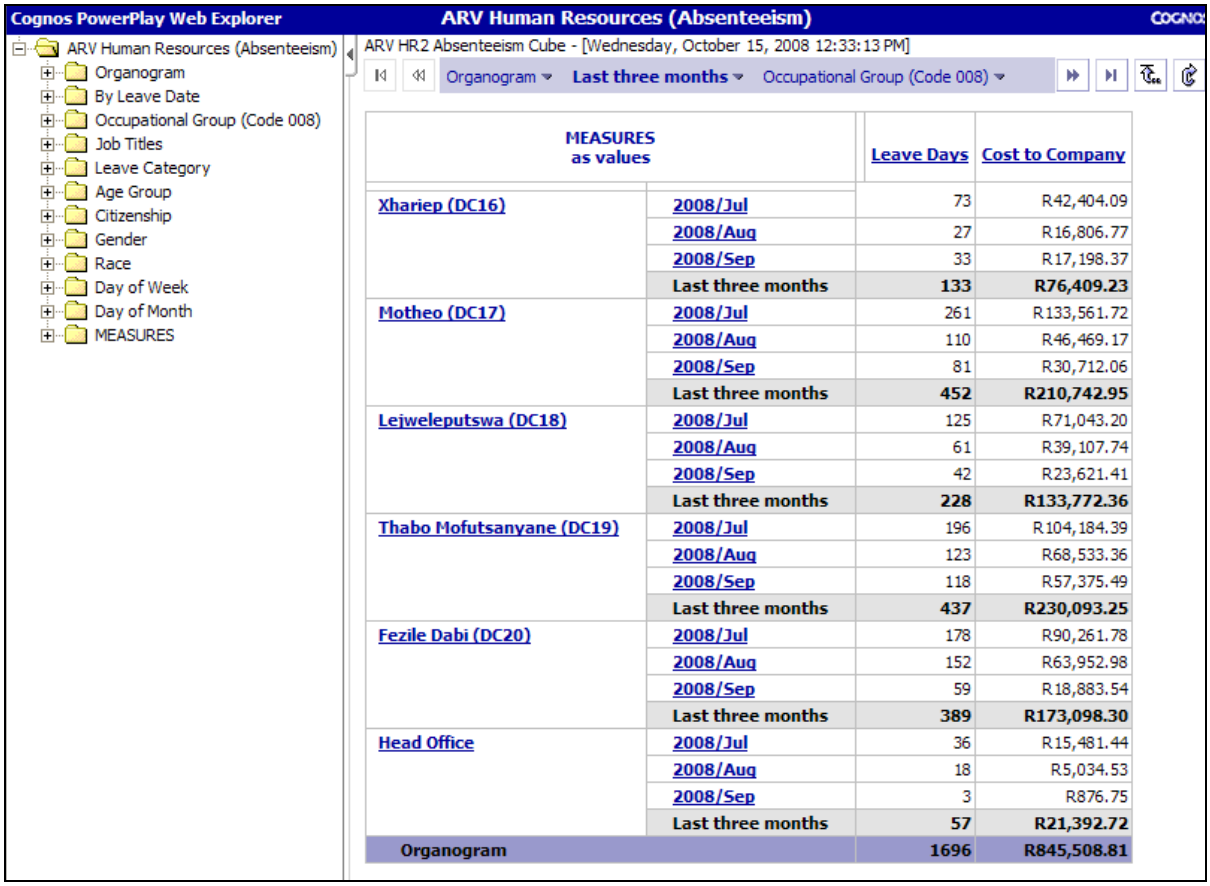
- Staff Establishment Cube (Antiretroviral based)
- Absenteeism Cube (Antiretroviral based)

Each individual cube focused on a particular subject area within HR and only included facilities involved in the ART programme. The reasoning behind a separate ARVHRDM was discussed in the previous chapter. Figures 7-6 to 7-7 provide screenshots of the listed cube's measures and dimensions as available in Cognos PowerPlay Web Explorer.



MEASURES as values		Approved Posts	Filled Posts	Bruto Salary	Vacancy Implications	Vacancy Rate
Xhariep (DC16)	2008/Jul	68	35	R487,751.76	R528,657.06	49%
	2008/Aug	68	34	R454,615.13	R567,936.33	50%
	2008/Sep	69	34	R546,918.61	R575,289.81	51%
	Last three months	**205**	**103**	**R1,489,285.50**	**R1,671,883.20**	**50%**
Lejweleputswa (DC18)	2008/Jul	84	56	R676,086.54	R341,476.34	33%
	2008/Aug	82	55	R848,951.67	R314,646.94	33%
	2008/Sep	82	56	R831,231.01	R289,817.74	32%
	Last three months	**248**	**167**	**R2,356,269.22**	**R945,941.02**	**33%**
Head Office	2008/Jul	28	15	R131,268.92	R189,978.27	46%
	2008/Aug	29	16	R63,139.22	R189,978.27	45%
	2008/Sep	29	15	R130,834.05	R189,978.27	48%
	Last three months	**86**	**46**	**R325,242.19**	**R569,934.81**	**47%**
Thabo Mofutsanyane (DC19)	2008/Jul	119	86	R1,004,053.97	R449,207.68	28%
	2008/Aug	119	88	R1,307,540.86	R402,574.93	26%
	2008/Sep	122	90	R1,442,704.92	R363,295.66	26%
	Last three months	**360**	**264**	**R3,754,299.75**	**R1,215,078.27**	**27%**
Fezile Dabi (DC20)	2008/Jul	104	59	R800,661.85	R675,551.58	43%
	2008/Aug	104	57	R629,509.62	R708,859.02	45%
	2008/Sep	113	54	R816,149.43	R698,688.82	52%
	Last three months	**321**	**170**	**R2,246,320.90**	**R2,083,099.42**	**47%**
Motheo (DC17)	2008/Jul	129	83	R1,082,463.23	R770,985.75	36%
	2008/Aug	130	83	R1,176,575.61	R772,278.01	36%
	2008/Sep	131	86	R1,180,467.49	R746,324.05	34%
	Last three months	**390**	**252**	**R3,439,506.33**	**R2,289,587.81**	**35%**
Organogram		**1610**	**1002**	**R13,610,923.89**	**R8,775,524.53**	**38%**

Figure 7-6: Staff Establishment Cube (Antiretroviral based)

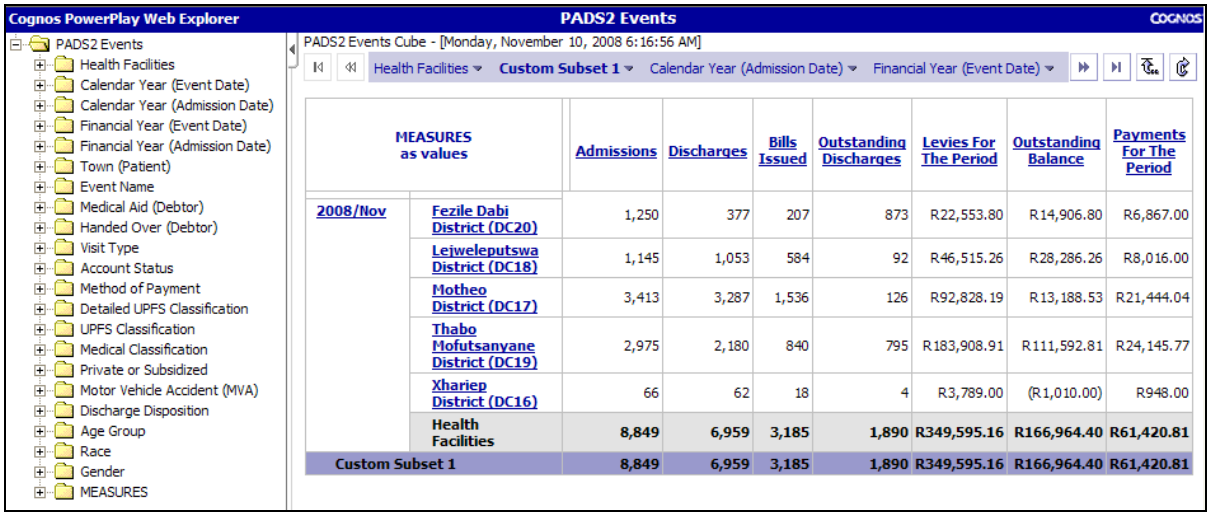


MEASURES as values		Leave Days	Cost to Company
Xhariep (DC16)	2008/Jul	73	R42,404.09
	2008/Aug	27	R16,806.77
	2008/Sep	33	R17,198.37
	Last three months	**133**	**R76,409.23**
Motheo (DC17)	2008/Jul	261	R133,561.72
	2008/Aug	110	R46,469.17
	2008/Sep	81	R30,712.06
	Last three months	**452**	**R210,742.95**
Lejweleputswa (DC18)	2008/Jul	125	R71,043.20
	2008/Aug	61	R39,107.74
	2008/Sep	42	R23,621.41
	Last three months	**228**	**R133,772.36**
Thabo Mofutsanyane (DC19)	2008/Jul	196	R104,184.39
	2008/Aug	123	R68,533.36
	2008/Sep	118	R57,375.49
	Last three months	**437**	**R230,093.25**
Fezile Dabi (DC20)	2008/Jul	178	R90,261.78
	2008/Aug	152	R63,952.98
	2008/Sep	59	R18,883.54
	Last three months	**389**	**R173,098.30**
Head Office	2008/Jul	36	R15,481.44
	2008/Aug	18	R5,034.53
	2008/Sep	3	R876.75
	Last three months	**57**	**R21,392.72**
Organogram		**1696**	**R845,508.81**

Figure 7-7: Absenteeism Cube (Antiretroviral based)

7.4.3. Patient Admissions and Debiting Data Mart (PADSDM)

For this data mart both a Cognos cube and ad-hoc query functionality was provided. Figure 7-8 provide a screenshot of the PADSDM cube's measures and dimensions as available in Cognos PowerPlay Web Explorer.



MEASURES as values		Admissions	Discharges	Bills Issued	Outstanding Discharges	Levies For The Period	Outstanding Balance	Payments For The Period
2008/Nov	Fezile Dabi District (DC20)	1,250	377	207	873	R22,553.80	R14,906.80	R6,867.00
	Lejweleputswa District (DC18)	1,145	1,053	584	92	R46,515.26	R28,286.26	R8,016.00
	Motheo District (DC17)	3,413	3,287	1,536	126	R92,828.19	R13,188.53	R21,444.04
	Thabo Mofutsanyane District (DC19)	2,975	2,180	840	795	R183,908.91	R111,592.81	R24,145.77
	Xhariep District (DC16)	66	62	18	4	R3,789.00	(R1,010.00)	R948.00
	Health Facilities	**8,849**	**6,959**	**3,185**	**1,890**	**R349,595.16**	**R166,964.40**	**R61,420.81**
Custom Subset 1		**8,849**	**6,959**	**3,185**	**1,890**	**R349,595.16**	**R166,964.40**	**R61,420.81**

Figure 7-8: PADS cube

7.4.4. Antiretroviral Clinical Data Mart (ARVDM)

For this data mart both a Cognos cube and ad-hoc query functionality was provided. Figure 7-9 provides a screenshot of a list of selected ARVDM ad-hoc queries that was created in Cognos Connection. Figure 7-10 provide a screenshot of the ARVDM cube's measures and dimensions as available in Cognos PowerPlay Web Explorer.

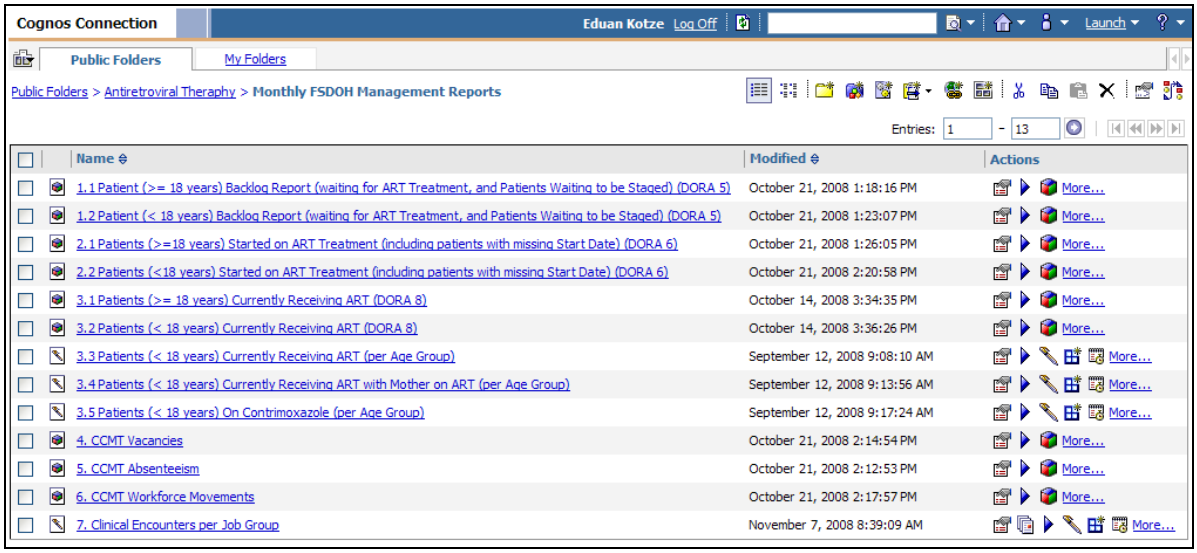


Figure 7-9: Cognos Connection reports for the ARVDM

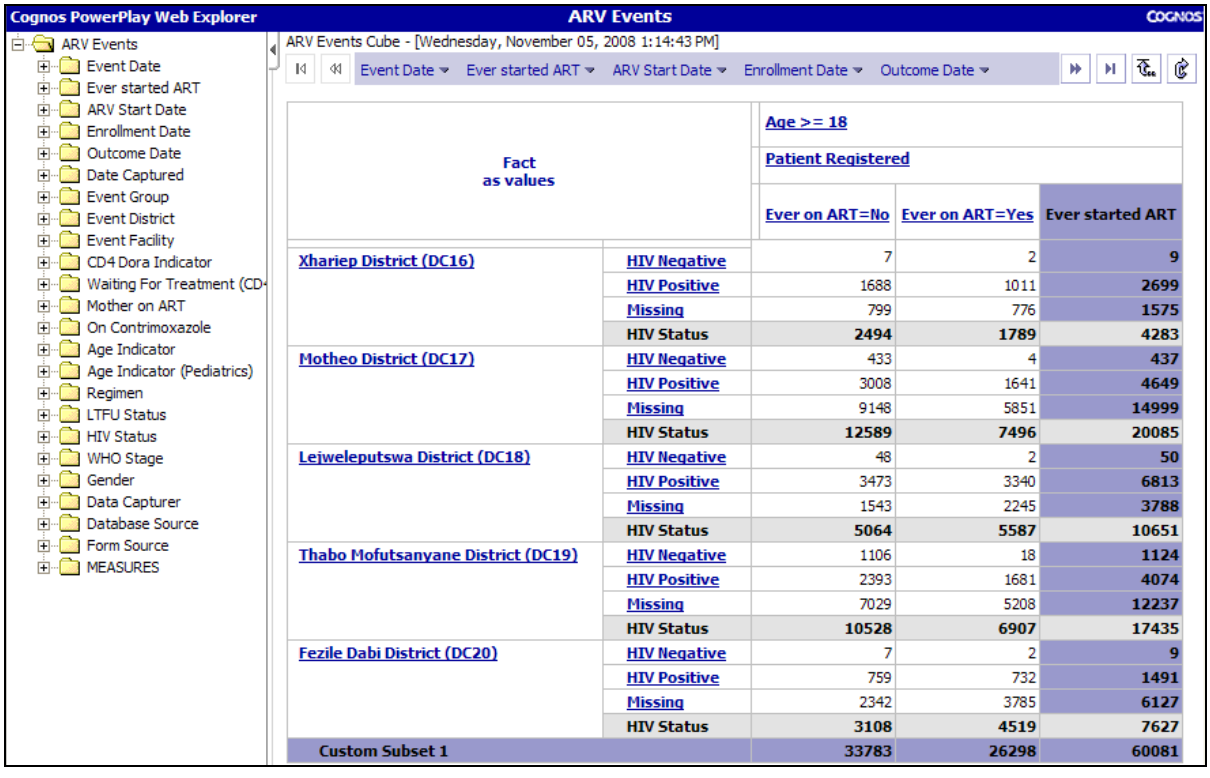


Figure 7-10: ARV Events Cube

7.4.5. Tuberculosis Data Mart (TBDM)

Ad-hoc query functionality was provided on the TBDM dimensional model. No cubes were requested or needed for this data mart.

7.4.6. Notifiable Diseases (NDDM)

Ad-hoc query functionality was provided on the NDDM dimensional model. No cubes were requested or needed for this data mart. Figure 7-11 provides a screenshot of a list of selected NDDM ad-hoc queries that was created in Cognos Connection.

Figure 7-11: Cognos Connection reports for the NDDM

7.5. Chapter Summary

This chapter provided the decisions and implementation methodology behind the FSDOH BI solution. Each data mart was covered and detail was provided on the cubes and ad-hoc query functionality that was provided for each one. The following chapter will now evaluate the data warehouse and business intelligence solution and test whether it was successful in providing the necessary strategic information for the managerial issues that was identified earlier by the relevant stakeholders.

CHAPTER 8 - EVALUATING THE BUSINESS INTELLIGENCE AND DATA WAREHOUSE SOLUTION

8.1. Introduction

The previous chapter provided an overview of the decisions that created the environment for transforming the data warehouse into a business intelligence (BI) solution. This chapter will focus on the *evaluation* phase of the action research cycle by evaluating the usage of the business intelligence and data warehouse solution and will determine if it was successful.

8.2. Survey Questions

Wixom and Watson (2001) investigated several implementation success factors affecting data warehouse success. Shin (2003) expanded on the work done by Wixom and Watson (2001) and examined the success factors in data warehousing by using system quality, information quality, service quality and user satisfaction as variables. The purpose of this study was to evaluate if the FSDOH data warehouse is used and whether it was addressing the relevant stakeholder's strategic information needs. Some of the questions of Shin (2003) were re-used in the questionnaire and new ones were added that were applicable to the FSDOH data warehouse evaluation.

The questionnaire consisted of three sections. Section 1 covered all the basic demographics of the respondents. Section 2 looked at how the respondents used the existing data warehouse while section 3 covered the perceptions of respondents on the Information from the data warehouse. See Appendix F for the layout and questions of the questionnaire.

8.3. Summary of Questionnaire Answers

8.3.1. Survey Data Collection

Data was collected from data warehouse users in the FSDOH which included ARV, human resources, revenue collection and hospital managers. All users received a questionnaire if they were either using the data warehouse themselves or request information from the knowledge workers who extract information for them from the data warehouse on a regular basis. The selected group was well represented over the five layers of employment at the FSDOH namely: production workers, supervisors, assistant managers, middle managers and top managers (see table 8-1). A total of 87 questionnaires were sent to this selected group of whom 51 responded. This translated into a response rate of 58.62%. Three (3) of the 51 respondents' questionnaires were very incomplete and discarded from the study. The final number of completed questionnaires to be used for analysis was 48.

For the purpose of this analysis the middle management and top management will be combined. 35.4% of the respondents were male while 64.6% were female. The FSDOH employs 16680 staff members according to their May 2008 employment statistics. The number of males were 5040 (30.22%) compared to 11640 (69.78%) females. The survey group therefore closely matches the demographics of the FSDOH. Table 8-1 summarizes the distribution of survey respondents in terms of their organizational status.

Table 8-1: Distribution of Respondents

Organizational Status	Number
Production Staff	13
Supervisor	10
Assistant Manager	13
Middle and Top Management	12
Total	48

8.3.2. Survey Data Analysis

A total of 64.8% of the survey respondents had a 3 year degree or higher, which indicated a high level of education among the respondents. Table 8-2 summarizes the distribution of the survey respondents in terms of their qualifications.

Table 8-2: Distribution of Qualifications

Organizational Status	Number	%
Senior Certificate	15	31.3
1 year diploma	2	4.2
2 year diploma	0	0
3 year diploma or degree	16	33.3
4 year B.Tech. or Honours degree or higher	15	31.3
Total	**48**	**100**

A total of 83.3% of the survey respondents indicated that they can help themselves with intermediate or expert computer tasks, while 95.7% indicated that they had more than 24 months of computer experience. Analytical skills are important when using a data warehouse and 84.4% of the survey respondents indicated that they had more than 24 months experience using an analytical tool such as Microsoft Excel.

Usage of the Data Warehouse

Most users (56.2 percent) accessed the system either monthly or quarterly. Six users used the system on a daily basis. Table 8-3 summarizes the frequency distribution of data warehouse access.

Table 8-3: Frequency Distribution of Data Warehouse Access

Data Warehouse Access	Frequency	%
Daily	6	12.5
Weekly	8	16.7
Monthly	16	33.3
Quarterly	11	22.9
Never / almost never	7	14.6

Types of tasks for which the data warehouse was used were also investigated. Eight main organizational tasks were included in the survey. The first four (decision-making support, status monitoring, planning, and forecasting) were considered more unstructured than the others (administration, accounting, resource allocation/budgeting and personnel management). Personnel management (62.5%) stood out as the most frequent task while forecasting (45.9%) where the least performed task.

For the unstructured tasks, most of the survey respondents would use the data warehouse sometimes or never while for the structured tasks the usage would be from very frequently to sometimes. According to the study done by Shin (2003), more users were using the data warehouse for unstructured duties rather than for routine or administrative responsibilities. For this study the weight tends to be for structured tasks instead of unstructured tasks. Table 8-4 summarizes the tasks and usage frequency.

Table 8-4: Task and Usage Frequency Distribution

% usage	Very frequently	Frequently	Sometimes	Rarely	Never
Decision-making support	10.3	23.1	28.2	10.3	28.2
Status monitoring	15.8	18.4	34.2	13.2	18.4
Planning	19.0	16.7	31.0	9.5	23.8
Forecasting	8.1	10.8	35.1	10.8	35.1
Administration	23.7	21.1	15.8	13.2	26.3
Accounting	13.9	16.7	27.8	8.3	33.3
Resource allocation	18.9	13.5	24.3	5.4	37.8
Personnel management	22.5	22.5	17.5	2.5	35.0

Finally, direct and indirect usage was also investigated. Most users (67.21%) indicated that they make use of either an assistant or knowledge workers at head office to obtain the information from the data warehouse for them. It is worth mentioning that the knowledge workers at head office were provided with certified training that was offered by Cognos South Africa. These courses empowered them to assist users with analysis requests. The remaining users (32.79%) retrieve the data by themselves and perform their own analysis. Most of these users attended an in-house business intelligence course, which introduced them to basics of Cognos reporting and Cognos cube analysis.

End-User Perceptions

Respondents were on the whole very positive about data quality, levels of detail and accuracy (see table 8-5). Most respondents (89.6%) agreed that that the data in the data warehouse is current enough to meet work needs. That was matched by 75% who disagreed that the data warehouse was out of date for a similar question that was negatively phrased. A total of 79.2% of the respondents indicated that the data warehouse maintains data at an appropriate level of detail to perform their tasks. This was matched by 68.1% who disagreed that the data warehouse does not have enough detail to make them more productive. Most respondents (70.3%) indicated that the data in the data warehouse is accurate and reliable and this was matched by (81.9) who were either unsure or disagreed that the data is inconsistent.

Table 8-5: Data quality, Levels of Details and Accuracy

% respondents	Agree Strongly	Agree	Unsure	Disagree	Disagree Strongly
Data quality - current	35.4	54.2	10.4	0	0
Data quality – out of date	2.1	6.3	16.7	37.5	37.5
Levels of detail -appropriate	35.4	43.8	16.7	2.1	2.1
Levels of detail – not enough detail	0	6.4	25.5	40.4	27.7
Data accurate and reliable	27.7	42.6	21.3	8.5	0
Data inconsistent	6.8	11.4	45.5	27.3	9.1

Next the data warehouse was evaluated in terms of functionality, flexibility, processing speed and ease of use. Respondents were on the whole very positive about these aspects (see table 8-6). A large number (78.9%) of respondents indicated that they were satisfied with the overall functionality of the data warehouse. This was matched by 80.4% of respondents who indicated that they were either unsure of did not agree that the data warehouse had no functional value to them.

Again a large number (68.8%) of respondents indicated they were satisfied with the overall flexibility of the data warehouse and that was matched by 68.7% who disagreed that they cannot perform their own analysis. A total of 68.7% of the respondents indicated that the data warehouse processing speed is good, but the negative question had a relative high number (41.7%) who indicated they were unsure. This could be due to the fact that network speed and bandwidth restrictions placed an uncertainty in the users' minds. A large number (79.2%) of respondents indicated that the data warehouse is convenient and easy to use, but interestingly 78.7% also indicated that more training is needed to find, understand and use the data warehouse.

Table 8-6: Functionality, Flexibility, Processing Speed and Ease of Use

% respondents	Agree Strongly	Agree	Unsure	Disagree	Disagree Strongly
Functionality	31.3	47.9	16.7	2.1	2.1
Functionality – no functional value	4.3	15.2	41.3	26.1	13.0
Flexibility	25.0	43.8	27.1	4.2	0
Flexibility – inability to perform own analysis	2.1	6.3	22.9	47.9	20.8
Processing speed	20.8	47.9	25	6.3	0
Processing speed – need to wait	4.2	18.8	41.7	29.2	6.3
Convenient and easy to use	25.0	54.2	14.6	6.3	0
Need more training	44.7	34	10.6	2.1	8.5

Finally, 93.8% of the respondents indicated that overall the data warehouse is a valuable asset for the FSDOH and it is recognized as being a critically important tool to improve the productivity of knowledge workers by providing strategic information.

8.4. Specifying Learning

The final phase of action research is to undertake *specifying learning*. There were two comments made by respondents that more work must be done to integrate patient information in the data warehouse. Follow up face-to-face interviews were held with these respondents to indentify and quantify the need. One of these respondents was an external researcher from the UCT Lung Institute, which is a research collaborator working with the FSDOH. It is important to note at this stage that permission was granted by the FSDOH to the UCT Lung Institute to use the data in the ARVDM to address unique and evolving research questions in HIV and AIDS currently unanswered by single cohorts. The comments made by the researcher on behalf of the UCT Lung Institute therefore carry significant value in evaluating the data warehouse. It was suggested to enrich the ARVDM with data from the other individual data marts and to make it as complete and integrated as possible. The newly identified need can be summarized as the requirement to provide an integrated and holistic view of antiretroviral therapy patients that will deliver longitudinal records containing the following data elements or facts:

- TB facts on patients that has a Lost-To-Follow-Up (LTFU) status and also enrolled on the TB Programme. These facts are recorded in the TBDM.

- Lung infections and diarrhoea related incidents for all ARV patients. Seasonal trends can assist to validate the ART treatment. Winter time it will be lung infections and summer time it will be diarrhoea. These facts are recorded in the NDDM.

- Hospital visits (non ARV related) that are not captured within the MPM system for all ARV patients. These facts are recorded in the In and Out Patient data from the Meditech and the PADS2 online databases.

In order to provide such a **longitudinal** record on a patient, the individual data marts will need to be linked together to form a cohesive clinical representation. It is envisaged that the ARVDM will play a central role in this linkage exercise and it will be enriched through linkage with patient records from all the individual data marts (TB Register, Notification Register, In/Out Patient data from Meditech and PADS, ART Data, Population Registry Data and NHLS data). The In/Out Patient Data Mart (HOSPDM) and NHLS Data Mart (NHLSDM) are two newly identified data marts and their construction will be discussed in later chapters.

It is envisaged that the linkage of records would dramatically reduce the number of queries fed back to data capturers at ART treatment and assessment sites, and permit monitoring beyond these centres (other primary care clinics, hospitals) to better estimate the unmet need for ART (numbers with eligible CD4 counts) and the effects of the programme on other programmes (e.g. TB) and health service utilisation (e.g. hospitalisation). Also, the longitudinal record could assist clinicians and

nurses that are in the process of enrolling ART patients or treating enrolled ART patients by providing a comprehensive longitudinal view of a patient.

8.5. Conceptualize the Proposed Longitudinal Record

It is important to conceptualize the requirement before the theory and mechanisms of constructing it can be discussed. The basic outline of the proposed longitudinal record is depicted below (see figure 8-1)

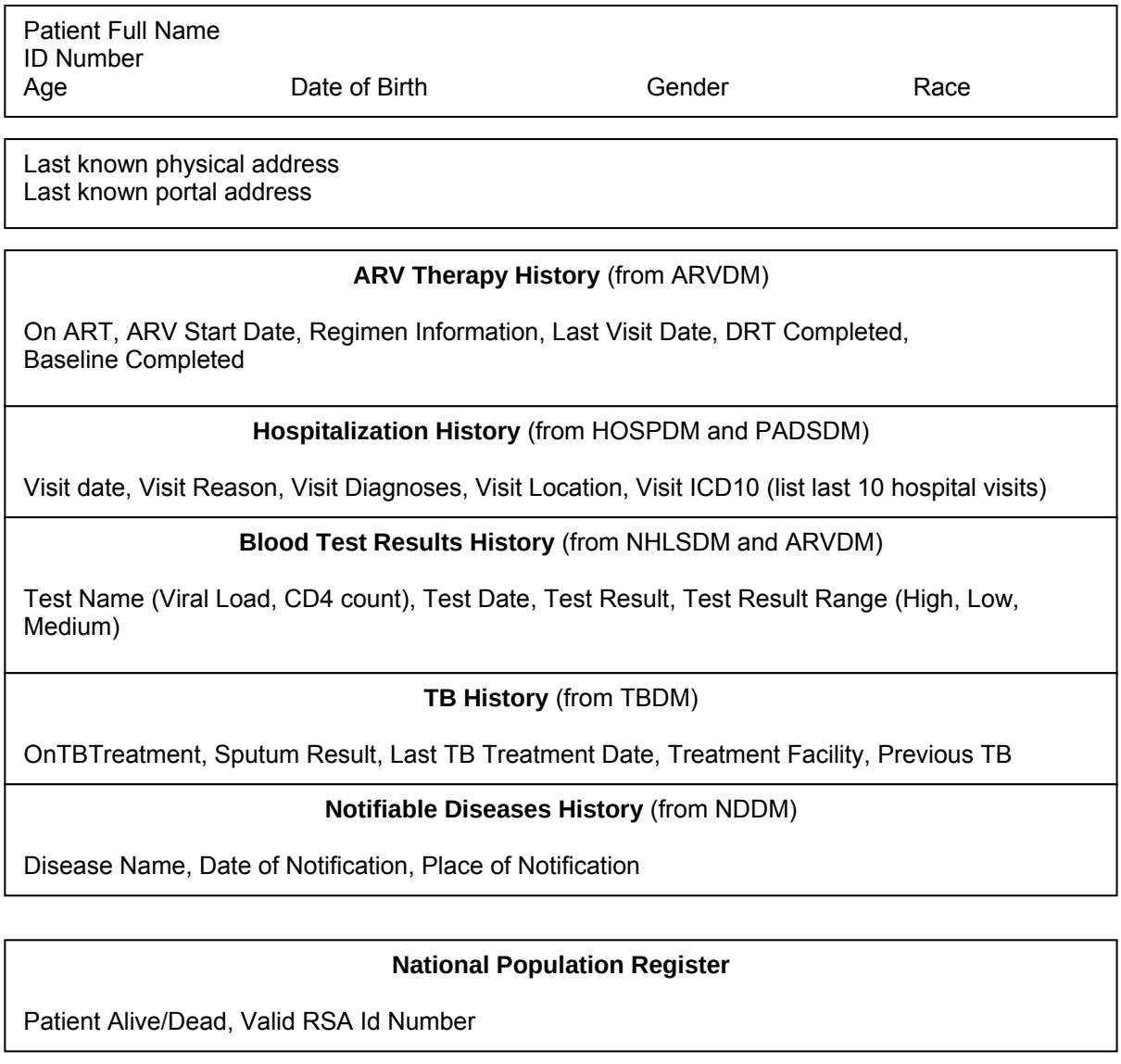


Figure 8-1: Conceptual view of a proposed longitudinal record

8.6. Chapter Summary

This chapter discussed the evaluation of the data warehouse and business intelligence solution and tested whether it was successful in providing the necessary strategic information that was identified earlier to the relevant stakeholders. From the findings one can conclude that the data warehouse is successfully used. New needs were identified and in particular the need to provide a longitudinal view of a patient. A basic outline and conceptual view of a proposed longitudinal record was provided and it was also indicated from which data marts each section will be extracted from. The following chapter will discuss the new needs that were identified and will document the integration of these data marts in the FSDOH data warehouse.

CHAPTER 9 - ADDITIONAL DATA MARTS

9.1. Introduction

The previous chapter covered the evaluation of the FSDOH data warehouse with all its individual data marts. Although very positive feedback was received, there was an outstanding yet critical need identified. This need was to provide an integrated and comprehensive view on all patient data related to the ARV programme. An *intervention* was required to add additional data marts and link up all these individual data marts and provide an **integrated longitudinal patient record.** This chapter will discuss the creation of the additional data marts and will also outline how it was integrated in the FSDOH data warehouse bus architecture. This chapter fits into the *problem diagnosis* phase of the action research cycle. Following chapters will discuss the background theory and methodology followed to link up all these individual data marts.

9.2. Additional Data Marts

The following additional data marts were identified to assist with providing a comprehensive and integrated view of patients on the ARV treatment programme

- NHLSDM (Blood results from National Health Laboratory Service database)
- HOSPDM (Meditech clinical encounters from the inpatients database)
- National Population Register at the Department of Home Affairs (DoHA)

The construction of each data mart will now be briefly discussed.

9.2.1. NHLS Blood Results Data Mart (NHLSDM)

This data mart would extract data from an external data source associated with the National Health Laboratory Service (NHLS). Key fields identified to be provided electronically from the national blood tests databank were the patient's personal details, all CD4 and viral load tests, test results and the date the tests were performed. It needs to be pointed out that this information is critical in determining the relevant ARV treatment regime for a patient. Without this information, it is almost impossible for a clinician to assess the patient's progress on the prescribed medication.

9.2.1.1. Addressing Managerial Outcomes

Prior to receiving the data electronically, the NHLS would send a paper fax (with the test results of each ARV encounter) to each treatment and assessment site. Data capturers would then capture these paper faxes with the test result into the Meditech MPM system. The end result led to incomplete or often incorrect CD4 and viral load data within the Meditech system. It was often the case, due to high workloads, that the NHLS blood results would not be captured at all and simply become a large pile of papers in a treatment or assessment site. To solve this challenge, electronic data was requested from the NHLS for incorporation into the FSDOH data warehouse.

9.2.1.2. Extraction, Transformation and Loading Challenges

The operational data source of the NHLSDM consisted of a single CVS (comma delimited) file, that contained basic patient demographics and the test (CD4 and Viral Load) results. When this chapter was written, only data for 2008 was received from the NHLS.

Constructing the NHLS_PATIENTS_DIM dimension was the most difficult task in populating the NHLSDM. During the loading process of the NHLS_PATIENTS_DIM dimension table it was discovered that only 8,908 of the 94,840 records had a valid unique patient identifier assigned to it. For these records the unique patient identifier was the RSA ID Number. No other unique identifier existed in the source file and a surrogate primary key was the only way forward. A surrogate primary key called Patient_Key was created and a "row number" value was assigned to it. It was not possible to create a de-duplicated patient dimension from the data file, and the end result was a one-to-one mapping between the fact table NHLS_FACT and the dimension table NHLS_PATIENT_DIM. Mechanisms to overcome this restriction will be covered in later chapters.

The NHLS data set followed its own location naming convention and had to be conformed to be part of the FSDOH data warehouse. The issue of standardizing the various versions of the same location name was discussed in section 6.4.7.1 (Conforming the Dimensions). The issue was resolved by introducing a single "helper" table called TREATMENT_LOCATION_BRIDGE with all the different

versions using an internal one-to-many map. The conformed dimension table TREATMENT_LOCATION_DIM was constructed from this TREATMENT_LOCATION_BRIDGE table and used in the PADSDM, ARVDM, TBDM and NDDM data marts. In order to make use of the standardized "helper" and conformed dimension tables, additional columns were added to the TREATMENT_LOCATION_BRIDGE table to accommodate the NHLSDM. See figure 9-1 for the layout of the modified TREATMENT_LOCATION_BRIDGE table.

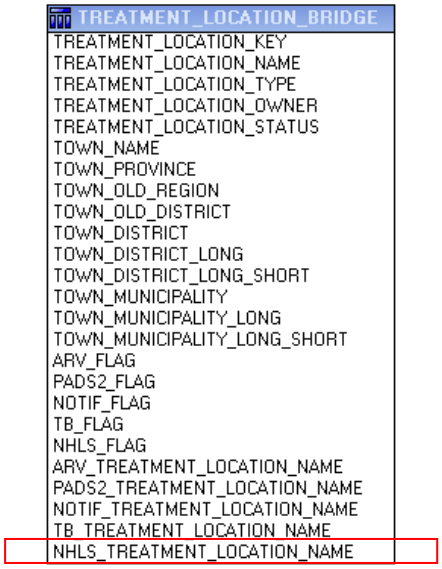


Figure 9-1: Modified TREATMENT_LOCATION_BRIDGE table

The *role-playing dimension* NHLS_LOCATION_DIM was added to the dimensional model as a database view from the TREATMENT_LOCATION_DIM dimension table. Referring to section 4.5.2.7, a role-playing dimension refers to a dimension that can play different roles in a fact table depending on the context. The *role-playing dimension* HNHLS_TESTDATE_DIM was added to the dimensional model as a database view from the DATE_DIM dimension table.

Finally the NHLS_FACT fact table was populated with CD4 tests and viral load facts and added to the dimensional model. The final dimensional model is depicted below in figure 9-2.

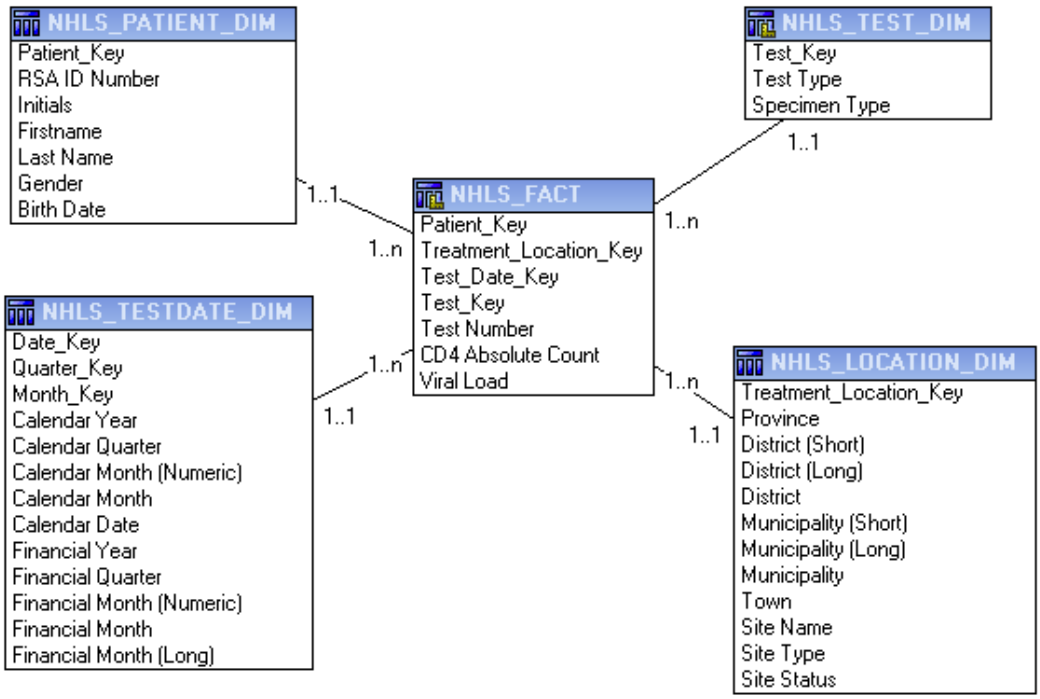


Figure 9-2: NHLSDM Dimensional Model

9.2.2. Hospitalization Data Mart (HOSPDM)

This data mart will contain clinical inpatient information on all patients admitted to any of the four hospitals using the larger Meditech HIS application. The hospitalization information will be extracted from the Meditech Admissions Module and the Meditech Outpatients Module (external data sources) and a data mart will be created based on clinical encounters.

9.2.2.1. Addressing Managerial Outcomes

By linking up all available hospital visit data on a patient, clinicians and professional nurses would be able to get a more comprehensive view on the treatment of a patient. Limited hospitalization data is already available from the PADSDM which covers 29 hospitals in the Free State Province. See Section 6.8 which covers the development of the PADSDM. This data mart is by no means complete and did not include any data from the remaining four hospitals that were using the larger Meditech HIS application. It was already pointed out during the data warehouse evaluation that hospitalization data was required from both systems to complete the **longitudinal** patient record. By creating a HOSPDM the challenge to provide a more comprehensive hospitalization view of a patient will be possible. The data from the HOSPDM combined with the data of the PADSDM will also produce a single result for the newly introduced DORA indicator that focuses on in-patient days of ARV patients.

9.2.2.2. Extraction, Transformation and Loading Challenges

The operational data source of the HOSPDM consisted of a two CVS (comma delimited) files. The one file provided all the hospital visit information while the other file provided the patient demographic details. During the loading process of the HOSP_MEDICAL_RECORD_FACT fact table, it came to light that no real fact could be identified. To avoid the situation of a factless fact table, a surrogate fact column called ***hospital_visit*** was inserted into the fact table. The value of the column was ***1*** for each row.

The *role-playing dimensions* HOSP_ADMISSIONDATE_DIM and HOSP_DISCHARGEDATE_DIM were introduced to the dimensional model as database views from the DATE_DIM dimension table. The *role-playing dimension* HOSP_TREATMENT_LOCATION_DIM was introduced to the dimensional model as a database view from the TREATMENT_LOCATION_DIM dimension table (with a filter extracting only the four Meditech HIS intuitions). The dimension tables HOSP_ACCOUNTS_DIM, HOSP_VISIT_DIM and HOSP_ICD10_DIM were added to complete the dimensional model. Finally a database foreign key was introduced between the ARV_PATIENTS_DEMO_DIM table of the ARVDM (see figure 9-3) and the HOSP_PATIENTS_DIM table using the common identifier "EPI Number". This foreign key enables the linkage of patients between the HOSPDM and ARVDM data marts. In Meditech, a patient received a unique number for each institution he/she visited but which is grouped together using an EPI Number and this identifier is often used to search for patients due to the lack of sufficiently completed RSA ID Numbers in both tables.

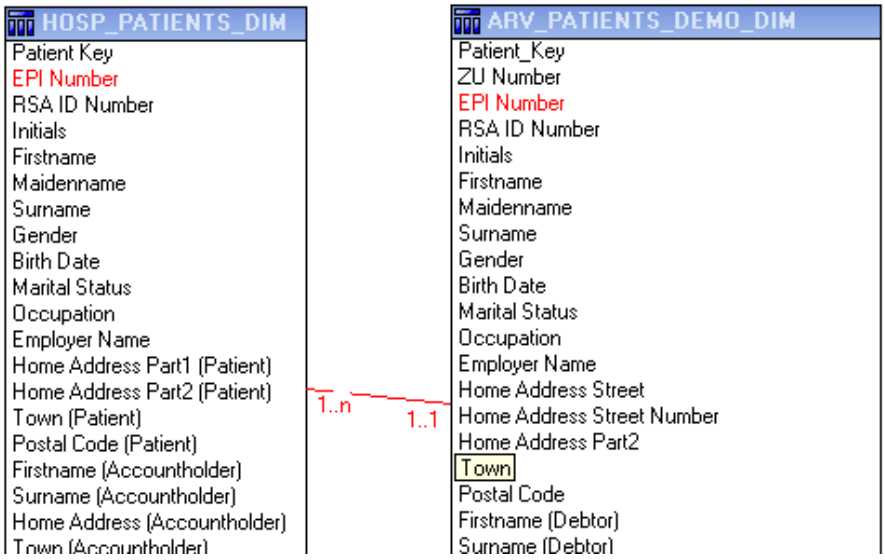


Figure 9-3: Illustrating the linkage between the ARVDM and HOSPDM

Tabulated in table 9.1 is an example of a patient that is on the ART Programme, and has also visited all four hospitals using the Meditech HIS in the Free State.

Table 9.1: Example of the EPI Number Grouping

EPI Number	Patient Key	Initials	Surname	Facility
00000002225	ZU0000231	B	Botha	***Any*** ARV Facility
00000002225	UH0004533	B	Botha	Universitas Hospital
00000002225	BM0004331	B	Botha	Boitumelo Hospital
00000002225	PMU0000231	B	Botha	Pelonomi Hospital
00000002225	GM0000121	B	Botha	Bongani Hospital

At the time of compiling this chapter the ARV_PATIENTS_DEMO_DIM table contained 62,947 rows and the HOSP_PATIENTS_DIM table contained 618,020 rows. A total of 618,010 (99.998%) of the HOSP_PATIENTS_DIM had an EPI Number assigned to the record. Only 59,744 (94.91%) of the rows in the ARV_PATIENTS_DEMO_DIM had an EPI Number. In theory this meant that 5.09% of the patients in the ARV Programme will not be linked with a hospital visit using traditional SQL joining techniques (primary-foreign key). Mechanisms to overcome this restriction will be covered in later chapters. The final dimensional model is depicted below in figure 9-4.

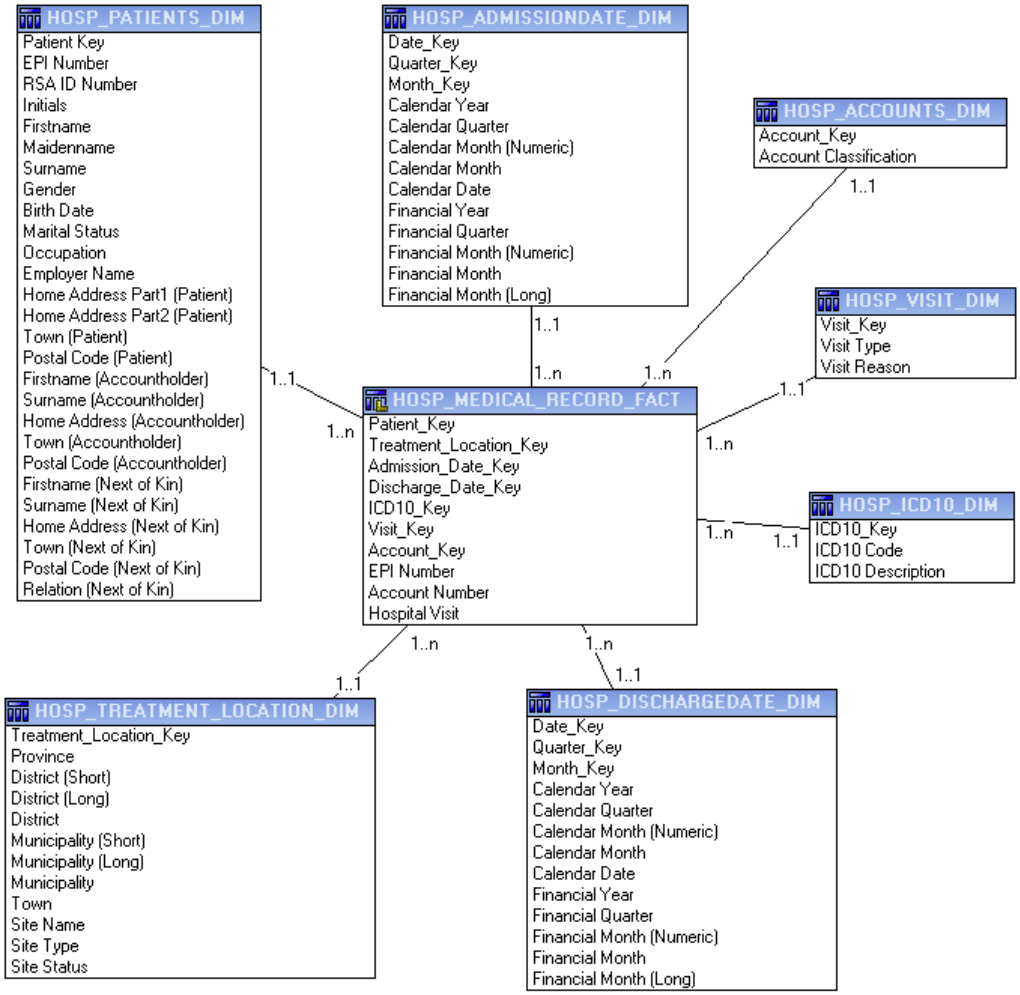


Figure 9-4: HOSPDM Dimensional Model

9.2.3. Linking up with the National Population Registry

This section will briefly describe the interface between the FSDOH Data Warehouse and the National Population Register situated at the Department of Home Affairs.

9.2.3.1. Addressing Managerial Outcomes

Internal Management Requirements:

Mortality numbers of patients on ARV therapy is a very important aspect to monitor. By linking the data provided from the National Population Register, the FSDOH was able to determine whether a patient was deceased or still alive and in care. If a patient mortality occurred while a patient is receiving ART, a follow-up investigation into the cause of the mortality had to be performed. This investigation was performed by specialist clinicians and the outcome was to establish whether it was linked to ARV treatment complications or due to secondary illness causes.

9.2.3.2. Extraction, Transformation and Loading Challenges

The data was provided on a monthly basis and loaded into the Oracle Data warehouse. No data mart was constructed for this exercise. The RSA ID Numbers of the ARV_PATIENT_DIM dimension table was compared with the RSA ID Number in the data provided from the National Population Register. If a match was found, the fields "***LTFU Status",*** "***LTFU Date", "Death LTFU Date", "Death Source", "Death Population Register" and "Death Population Register Date"*** were updated to reflect the newly obtained intelligence in the ARV_PATIENT_DIM dimension table (see figure 9-5).

Figure 9-5: Partial definition of the ARV_PATIENT_DIM dimension table

9.3. Chapter Summary

This chapter described the creation of the additional data marts that was identified during the data warehouse evaluation study. The *intervention action* to link up all these data marts is still outstanding and will be addressed using record linkage methods. The following chapter will outline the theory associated with record linkage to provide the basis of linking data marts together in order to make a longitudinal record possible.

CHAPTER 10 - GENERAL PRINCIPLES OF RECORD LINKAGE

10.1. Introduction

The purpose of this chapter is to provide the background knowledge for the forthcoming chapters on the usage of record linkage methods. The first section will expand on the background and objective of record linkage. The different record linkage methods will then be discussed, followed by an in depth look at probabilistic record linkage. The chapter will conclude with a proposed linkage solution for the FSDOH data warehouse and how it differs from work already done in the healthcare field. This chapter fits into the *action planning* phase of the action research cycle.

10.2. Background

Record linkage is the process of combining information about an individual, facility or entity residing in one or more databases (Grannis, Overhage and McDonald, 2004) or from a variety of computerized files (Winkler, 2005). Record linkage is also called *object identification*, *data cleaning*, *approximate matching* or *approximate joins*, *fuzzy matching* and *entity resolution* (Winkler, 2005). The basic methods compare names and address information across pairs of files to determine those pairs of records that are associated with the same entity (person, business unit, etc.). Because much of the data is based on textual information such as names and addresses, most of the advances in record linkage methods have been in the computer science literature (Winkler, 2005).

Individual record linkage involves usually at least two files. For this discussion two files A and B will be matched. Each file consists of a fixed number of fields, which contains information to be matched between A and B. Obviously, both files need to have a number of **equivalent** fields for Record Linkage to work. The objective of the record linkage process is to classify each pair as belonging to one of two sets: the set of matched record pairs M and the set of unmatched record pairs U. If file A contains 2000 records and file B contains 2000 records there are 4,000,000 possible record pairs, but only 2000 possible matches (if there are no duplicates in the files). Therefore, set M can contain 2000 pairs and set U can contain the remaining 3,998,000 pairs.

10.2.1. Blocking

The mathematical approach to record linkage theory becomes more complicated when allowing for blocking or missing data (Clark, 2004). For any files of reasonable size it is not feasible to compare all record pairs since the number of possible pairs is the product of the number of records in each file (Jaro, 1995) and it becomes impractical when the files are large (Clark, 2004).

To put this problem in perspective, let us examine a database containing *n* records. Record linkage is often described as an O(*n*2) complexity problem, due to its Cartesian product aspect (Sauleau, Paumier and Buemi, 2005). Should one compare each record to every other record, it will require (*n*2-*n*)/2 comparisons. Using this approach will be the most reliable (as no record is missed during the comparison process) but it is also the most time consuming and least effective (CPU load speaking) (Sauleau et al., 2005). In order to reduce the number of comparisons, indexation of databases by blocking techniques is used. In theory this means that the data sets are split into smaller blocks and only records within the same blocks are compared. Take for example the 4,000,000 possible record pairs in the previous section. This is a very challenging exercise to link only 2000 records with another 2000 records. Imagine 100,000 records compared with 200,000 records (20,000,000,000 or $2x10^{10}$), which is still considered a small to medium-sized file. In order to reduce the number of comparisons, indexation of databases by blocking techniques is used. In theory this means that the data sets are split into smaller blocks and only records within the same blocks are compared.

To understand the concept of blocking, take the example of the field age with 100 possible ages. If there are 100 possible ages, then this field partitions a file into 100 subsets. The first subset would be people with an age of 0, the next is those with an age of 1 and so on until those with an age of 100. These subsets are called blocks (Jaro, 1995). If it is assumed that the age in the set M (2000 pairs) and set N (2000 pairs) are distributed uniformly, there would be 20 (2000 pairs / 100 ages) records for people of age 0 in each file, 20 for age 1, 20 for age 2 and so on.

The pairs of records to be compared are taken from records in the same block. The first block would consist of all persons of age 0 in set M and set N. This would be (2000/100) x (2000/100) = 20 x 20 = 400 record pairs. The second block would consist of all persons in set M and set N with an age of 1. Again this would be 400 pairs. If this process is completed, 40,000 pairs would have been compared, instead of the 4,000,000 possible record pairs without blocking.

The obvious problem with this approach is the fact that only records having the same value in the blocking field (M_{age} = N_{age} = 1) will be compared. It also means that records not matching on the blocking fields will automatically be classified as non-matched. In fact, if age was in error in one of the fields, then the records involved are considered non-matched. To get around this problem, multiple passes are used (Jaro, 1995).

10.2.2. Multiple Passes in Blocking

The idea of multiple passes is when records do not match on blocking field 1 (age) in pass 1, it can be re-matched using another blocking field, for example postal code of residence in pass 2. Then it will be only those cases that have errors on both the age and postal code fields that will be non-matched. If this is a problem, a third pass can be run. Errors on all three blocking fields are unlikely (Jaro, 1995). To maximize the idea of ***multiple passes***, fields should be used for blocking fields that

are with the most number of distinct values possible and highest reliability. For example gender is a bad example, since it will divide the file only into two subsets (male and female). The blocking strategies for each pass should be independent to the extent possible (Jaro, 1995). For example, if a pair of files have surname, firstname, gender and birthdate (year, month, day) fields, then the first pass could be blocked on surname, gender and birth year. The second pass could be blocked on birth month, birth day and firstname. Errors in surname, for example, would be unmatched in pass 1, but would be likely to be matched in pass 2.

10.2.3. Error Rates

In record linkage, the probability of falsely matching records that should not have been matched must be balanced against the probability of failing to match records that should have been matched. Records that are falsely matched (‘‘mismatches’’ or ‘‘homonym errors’’) will lead to misidentification of the outcome for specific cases as well as underestimation of the total number of cases; records that are falsely unmatched (‘‘false non-matches’’, ‘‘erroneous non-matches’’, ‘‘failures to match’’, or ‘‘synonym errors’’) will lead to missing data from one or the other source and overestimation of the total number of cases. The theoretical magnitude of these errors can be estimated algebraically using certain assumptions. The frequency of false positives and false negatives can be expressed in familiar terms of sensitivity, specificity, positive predictive value (PPV), and negative predictive value (NPV), as laid out in table 10.1.

Table 10.1: Possible outcomes for two records from different files

	Records truly are from the same person	Records truly are ***not*** from the same person
Records matched	Truly matched (TM)	Falsely matched (FM)
Records ***not*** matched	Falsely unmatched (FU)	Truly unmatched (TU)

Sensitivity = TM / (TM+FU)
Specificity = TU / (TU+FM)
Positive predictive value (PPV) = TM / (TM+FM)
Negative predictive value (NPV) =TU / (TU+FU)

In practice, the number of records truly unmatched is generally so large that specificity and NPV are not useful measurements. Furthermore, for any real application, it may be difficult to specify the ‘‘gold standard’’ against which matched or unmatched records are considered ‘‘true’’ or ‘‘false’’. Because of this, a method to estimate the PPV value based on the frequency of duplicate links has been proposed (Clark, 2004).

10.2.4. Weights

From the previous discussion it can be concluded that each field will have two probabilities associated with it. The *m probability* is the probability that a field agrees given that the record pair being examined is a match pair. The *u probability* is the probability that a field agrees given that the record pair being examined is an unmatched pair. The weight for a field is computed as the logarithm to the base two of the ratio of *m* probability and *u* probability.

To illustrate this, the previous example of file A and B, each containing 2000 records can be used. Set M contains 2000 pairs and set U contains the remaining 3,998,000 pairs. Set *M* is the set of ***true matches*** (Winkler, 2005) and set *U* is the set of ***true non-matches*** (Winkler, 1999, 2005). Assume that gender has a 10% error rate and the RSA Identity Number has a 40% error rate. The m probability is 0.9 and the u probability is 0.5 in situations with an equal number of males and females. The weight for gender would be:

$Log_2(m/u) = ln(m/u/ln(2) = ln(0.9/0.5)/ln(2) = 0.85$

Now take the RSA Identity Number with *m* given as 0.6 and the probability that a match is 1 in 10 million (*u = 0.0000001*). The weight for RSA Identity Number would be 22.51. The weights have now captured what is known intuitively about the fields (Jaro, 1995). For each record pair compared, a composite may be computed as the sum of the individual weights for all field comparisons. If a field agrees in the pair being compared, the agreement weight, as computed above is used. If a field disagrees, the following weight formula is used. $Log_2[(1-m)/(1-u)]$. This results in a negative weight. Thus, agreements **add** to the composite weight and disagreements **subtracts** from the composite weight. Obviously, the higher the score, the greater the agreement (Jaro, 1995).

10.3. Computerized Record Linkage Approaches

There are three commonly used computerized record linkage approaches that exist today: **Match-Merge** (Cole, 2003), **Deterministic record linkage** and **Probabilistic record linkage** (Winkler, 2005). Each one of the three approaches will now be briefly discussed to provide the necessary theoretical background.

10.3.1. Match-Merge Record Linkage

Match-merge techniques are generally used only when information originates from the same data system or when identifiers (such as SSN) are very reliable (Cole, 2003). A **match-merge** relies on an exact match of a *single* common identifier present in two files (Cole, 2003).

10.3.2. Deterministic Record Linkage

Deterministic record linkage requires an exact match of identifying information, but uses *multiple* criteria to establish a match (Cole, 2003). The method generates links (*Li, Quan, Fong and Lu, 2006) or a set of rules (Grannis et al., 2004) on the basis of a full agreement of a unique identifier or a set of common identifiers. This minimizes the uncertainties in the match between two databases since only a complete match on a set of personal variables is accepted at the cost of lowering the linkage rate (Li et al., 2006).

According to Gu, Baxter, Vickers and Rainsford (2003) deterministic record linkage also assumes error-free identifying fields and links records that exactly match on these identifying fields. For large multiple data sources, shared error-free identifying fields are uncommon. When no error-free unique identifier is shared by all of the data sources, a probabilistic record linkage technique can be used to potentially join or merge the data sources. This will be discussed in section 10.3.3.

10.3.2.1. Deterministic Record Linkage in Healthcare

In a research study performed by Muse, Mikl and Smith (1995), record-linkage algorithms were used between an **anonymous** statewide registry of all AIDS cases and a statewide hospital discharge file. Personal names were absent from both files. The study used combinations of the patient's date of birth (DOB), gender, and hospital identification code (PFI) and corresponding dates of hospitalization.

Liu and Wen (2000) did a study using the linkage technology in the SAS software for Unix (version 6.12) to link hospital discharge data and neonatal data. They argued that to allow an examination of the relation between length of newborn hospital stay at birth and subsequent neonatal readmission, a linkage of readmission records with the infant's own birth record is required. Data from both birth and

readmission records were used and the following matching variables were used: Province, Institute Number, Health care number, Postal Code, Residence Code, Date of Birth, Sex, Admission date and Discharge date. The study concluded that a *deterministic matching algorithm* is a feasible and convenient approach to data linkage for the study of neonatal readmission.

Li et al. (2006) showed how the deterministic record linkage approach was used to link three administrative health databases. Their findings suggested that deterministic record linkage using three basic indicators (i.e., surname, sex and date of birth) appeared to generate the highest linkage rate among three commonly used databases (population registry, hospital discharge data and vital statistics registry) in health service research. Li et al. (2006) also stated that surname, first name, sex, and date of birth were less likely to be changed over time, compared to other identifiers like address. For this reason, the address identifier was not used in the study.

10.3.3. Probabilistic Record Linkage

Probabilistic record linkage identifies a match between records based on a formal statistical model and a linkage is made based on a calculated statistical probability (Cole, 2003). A probability is used to determine whether a pair of records approximately refers to the same individual (Li et al., 2006) and when the probability of a match exceeds a certain threshold, the linkage is made (Cole, 2003).

The advantage of probabilistic record linkage is that it uses all available identifiers to establish a match (e.g., name, sex, date of birth, SSN, race, address, phone number) and does not require identifiers to match exactly. Identifiers that do not match exactly are assigned a “distance” measure to express the degree of difference between files. Each identifier is assigned a weight and the total weighted comparison yields a score, which is used to classify records as linked, not linked, or uncertainly linked according to whether the statistical probability of a match exceeds a certain threshold (Winkler, 1999).

Probabilistic linkage software will typically declare a “link” or “potential link” (Howe, 1998) for record-pairs with high match likelihood scores and will declare a “non-link” if the score is very low. It requires a human operator to evaluate the record-pair when the computed likelihood is within an indeterminate middle range (Grannis et al., 2003). When two records truly refer to the same individual, they are called “matched” (Howe, 1998). See figure 10-1 for a typical two-threshold scheme for probabilistic scores using human review. Record pairs between the upper and lower thresholds are manually reviewed for true- or false-link status.

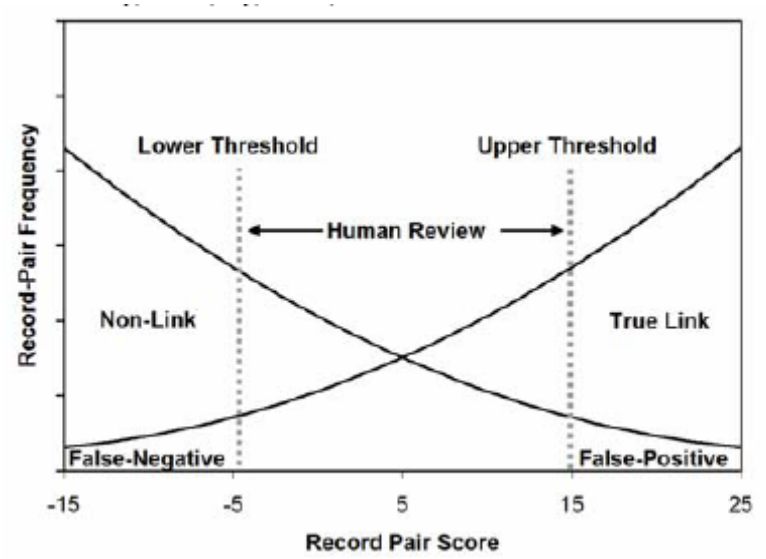


Figure 10-1: Typical two-threshold scheme for probabilistic scores using human review

Correct linkage depends on the amount of personal identifying information available on the records being linked. When enough information is available to use deterministic record linkage, this method should increase the correct linkage with little sacrifice in lowering the linkage rate. On the other hand, when unique personal identifiers (such as social insurance number or personal health number) are not available, the linkage and correct linkage rates depend heavily on the uniqueness of a set of proxies (such as name, sex and date of birth) (Li et al., 2006).

10.3.3.1. Probabilistic Record Linkage Implementations

Probabilistic record linkage has been implemented in record linkage software systems that are available commercially and from government agencies (Cole, 2003). Most of the software systems available commercially have been developed in the USA and Canada. Examples of commercially software systems include the following: AutoMatch Software (Jaro, 1995) now part of Ascential Software (Cole, 2003), ChoiceMaker (Cole, 2003) and SAS (Cole, 2003). An example of a software system available in government agencies is the Generalized Record Linkage System (GRLS) which was developed by Statistics Canada (Fair, 2004) in response to business and health needs in the public sector. According to Fair (2004), the current version of GRLS is version 4 and it runs in a client-server environment with Oracle and a C Compiler.

10.3.3.2. Probabilistic Record Linkage in Healthcare

Bernillon, Lievre, Pillonel, Laporte and Costagliola (2000) performed a research study in France in 2000 on the record-linkage of **two anonymous databases** to estimate the completeness of the French mandatory AIDS surveillance system over the 1990-1993 period. The authors reported in their study that the record linkage procedure used the following information: date of birth (month and year),

sex, HIV infection risk category, date of AIDS diagnosis, date of last visit to a CISIH, the AIDS-defining clinical manifestations (maximum of eight), the 'département' of declaration, and date of death for the 5963 known to have died. The 'département' of declaration is the 'département' of the hospital where the patient is treated at the moment of his transition from HIV-positive to AIDS status. It was *unclear* from their journal article as to which record-linkage algorithm was used and instead it stated that the two databases shared no common identifier, and an algorithm was developed in order to cross-match the two databases automatically. This algorithm comprised of two steps: (1) determination of potential links and (2) identification of the real matches among these potential links. The algorithm was successful in the consolidation of the two surveillance systems and resulted in the identification of 91.4% of the total number of cases. The algorithm was implemented using the SQL relational database language and the data was integrated in an Oracle database (Bernillon et al., 2000).

Grannis et al. (2003) employed an estimator function using the Expectation Maximization (EM) algorithm to establish a single true-link threshold. This was done to avoid human review of record-pairs when the computed likelihood is with an indeterminate middle range (between high match record-pairs and low-match record pairs) using a probabilistic linkage technique. The authors concluded that the EM algorithm estimated linkage parameters without human intervention and the methodology may be used where record linkage is possible, but human intervention is not possible or practical. The algorithm used two separate 6,000 record pair files from two hospital registries.

Nitsch, Morton, DeStavola, Clark and Leon (2006) have showed that probabilistic methods can be used to link records in a cohort study. This linkage was based on five fields: surname, maiden name, first and second initial, full date of birth and current postcode. The probability matching methodology used has been formally described and is based on the approaches developed by Newcombe and others in Canada and employed by the Oxford Record linkage study (Nitch et al., 2006). Because of variation and inconsistency in spelling of surnames two phonetic coding systems were used that converted phonetically similar surnames to the same linkage key value. These were the ***NYSIIS*** (New York State Identification and Intelligence System) and ***Soundex*** codes (adapted to Scottish surnames). The linkage process involved the computation of a score for each potential match. The value of this score depended on the probability that each of the five fields agreed. To work this out, weights were computed corresponding to each field. Each weight was defined as the log (to base 2) odds in favor of a match between two records according to that field (Jaro, 1995). Its value depended on how many possible values the field could take, as well as how much misspelling occurred. Records were classified as achieving an **acceptable match** if the sum of the five weights (the 'linkage score') was greater than a specified value (Nitch et al., 2006).

10.4. String Comparison Mechanisms

In this section the Fellegi-Sunter model of record linkage, name and address standardization, phonetic compression and string comparators for approximate string comparison record linkage methods will be described.

10.4.1. Fellegi-Sunter Model of Record Linkage

Fair (1997) states that the initial definition of record linkage originated by H.L. Dunn in 1946. The ideas of modern record linkage originated with geneticist H. Newcombe who introduced a formal mathematical model (Winkler, 2005; Cole, 2003), using odds ratios of frequencies and the decision rules for delineating matches and non-matches (Winkler, 1999). The first part of Newcombe's approach was that the relative frequency of the occurrence of a value of a string such as a surname among matches and non-matches could be used in computing a binit weight (score) associated with the matching of two records. The second part was that the scores over different fields such as surname, first name, age, etc. could be added to obtain an overall matching score (Winkler, 1999).

More specifically, Newcombe considered odds ratios:

$$\log_2(p_L) - \log_2(p_F) \quad (1)$$

where p_L is the relative frequency among links and p_F is the relative frequency among non-links. Since the true matching status is often not known, he suggested approximating the above odds ratio with the following ratio (Winkler, 1999):

$$\log_2(p_R) - \log_2(p_R)^2 2 \quad (2)$$

where p_R is the frequency of a particular string (first initial, birthplace, etc.). If one matches a large universe file with itself, then the second ratio is a good approximation of the first ratio (Winkler, 1999).

According to Winkler (1999) and Cole (2003), Fellegi and Sunter (1969) provided the formal mathematical foundations of record linkage. Their theory demonstrated the optimality of the decision rules used by Newcombe et al. (1959) and introduced a variety of ways of estimating crucial matching probabilities (parameters) directly from the files being matched (Winkler, 1999). Fellegi and Sunter made use of concepts introduced by Newcombe et al. (1959) and considered ratios of probabilities of the form (Winkler, 2005):

$$R = P(\gamma \in \Gamma \mid M) / P(\gamma \in \Gamma \mid U)$$

In this expression γ is an arbitrary agreement pattern in a comparison space Γ. For instance, Γ might consist of eight patterns representing simple agreement or not on the largest name component, street name, and street number. Alternatively, each $\gamma \in \Gamma$ might additionally account for the relative frequency with which specific values of name components such as "Smith", "Zabrinsky", "AAA", and "Capitol" occur. The ratio R or any monotone increasing function of it such as the natural log is referred to as a matching weight (or score) (Winkler, 2005).

10.4.2. Name and address standardization

Standardization is a rule-based mechanism and consists of replacing various spellings of words with a single spelling. For instance, different spellings and abbreviations of 'Incorporated' might be replaced with the single standardized spelling 'Inc.' (Winkler, 2005).

Successful standardization exercises were performed for the US Census of Agriculture matching system. Promising new methods based on *Hidden Markov* models may improve over the rule-based name standardization. Although the methods are clearly an improvement over more conventional address standardization methods for difficult situations such as Asian or Indian addresses, they did not perform as well as more conventional methods of name standardization (Winkler, 2005).

10.4.3. Phonetic Compression

Phonetic encoding algorithms are used to minimize variations in spelling of what are effectively the same attribute for example surname or firstname. There are several well-known phonetic compression algorithms; examples include Metaphone, New York State Identification and Intelligence System algorithm (NYSIIS) (Roos, Wajda and Nicol, 1986) and Soundex (Grannis et al., 2004).

The NYSIIS algorithm has 11 basic rules that replace common pronunciation variations with standardized characters, removes repeated characters, and replaces all vowels with the letter 'A'. Because it retains information on the sequence of vowels, NYSIIS has higher discriminating power than Soundex. The NYSIIS transformations of 'TAMMIE' and 'TAMMY' are both 'TANY' (Grannis et al., 2004).

10.4.4. String comparators

In many matching situations, it is not possible to compare two strings exactly (character-by-character) because of typographical errors (Winkler, 1999, 2005). An algorithm is required to assist with these comparisons while taking the typographical errors into account. Listed below are string comparison algorithms which include the ***Jaro and Jaro-Winkler algorithm, Longest Common Substring (LCS)*** algorithm and the ***Levenshtein Edit Distance (LEV)*** algorithm.

10.4.4.1. Jaro and Jaro-Winkler

Jaro introduced a string comparator that accounts for insertions, deletions, and transpositions (Winkler, 1999) called the ***Jaro*** algorithm (Winkler, 2005). The basic Jaro algorithm has three components: (1) compute the string lengths, (2) find the number of ***common*** characters in the two strings, and (3) find the number of ***transpositions***. The definition of common is that the agreeing character or corresponding characters (Grannis et al., 2004) must be within half the length of the shorter string. The definition of ***transposition*** is that the character from one string is out of order with the corresponding common character from the other string. In other words a *transposition* occurs when the order of corresponding common characters is reversed (Grannis et al., 2004).

According to Winkler (1999), the string comparator value (rescaled for consistency with the practice in computer science) is as follows:

$$\Phi j(s_1, s_2) = 1/3(N_C/len_{s1} + N_C/len_{s2} + 0.5N_t/N_C)$$

s_1 and s_2 are the strings with lengths len_{s1} and len_{s2}, respectively, N_C is the number of common characters between strings s_1 and s_2 where the distance for common is half of the minimum length of s_1 and s_2, and N_t is the number of transpositions. The number of transpositions N_t is computed somewhat differently from the obvious manner.

The ***Jaro*** string comparator was modified when Winkler (1990) showed how a variant of the Jaro string comparator Φ dramatically improves matching efficacy in comparison to situations when string comparators are not used. From this work the ***Jaro-Winkler*** string comparator was produced (Winkler, 2005). According to Cohen, Ravikumar and Fienberg *(*2003b), the *Jaro* string comparator is a metric widely used in the record-linkage community, with and without a variation due to Winkler. The ***Jaro*** and later extended ***Jaro-Winkler*** string comparator will be used in this study as research has shown it to be a fast heuristic scheme and a good distance metric (Cohen et al., 2003a). The comparator score for 'TAMMY SHACKELFORD' and 'TAMMIE SHACKLEFORD' is 0.9442 (Grannis et al., 2004) using the Jaro-Winkler string comparator.

10.4.4.2. Longest Common Substring (LCS)

This algorithm generates a nearness metric by iteratively locating and deleting the longest common substring between two strings (Grannis et al., 2004). The substrings must meet a minimum length requirement, which is set to three for our analysis. The nearness metric is calculated by dividing the total length of the shared substrings by the length of the shorter of the two strings being compared. For example, the LCS score for the names 'TAMMY SHACKELFORD' and 'TAMMIE SHACKLEFORD' is calculated as follows: The total length of the common substrings is [5 (SHACK) + 4 (TAMM) + 4 (FORD)] = 13. The length of the shorter name string (ignoring white space) is 16, therefore the LCS score is (13÷16) = 0.8125

10.4.4.3. Edit-Distance Functions

Distance functions map a pair of strings *s* and *t* to a real number *r*, where a smaller value of *r* indicates greater similarity between *s* and *t*. *Similarity functions* are analogous, except that larger values indicate greater similarity. At some risk of confusing the reader, these terms will be used interchangeably, depending on which interpretation is most natural. One important class of distance functions are *edit distances*, in which distance is the cost of the best sequence of *edit operations* that convert *s* to *t.* Typical edit operations are character insertion, deletion, and substitution, and each operation much be assigned a cost *(*Cohen, Ravikumar and Fienberg, 2003a).

Edit Distance uses dynamic programming to determine the minimum number of insertions, deletions, and substitutions to get from one string to another. The **Bigram** metric counts the number of consecutive pairs of characters that agree between two strings (Winkler, 2005).

- ***Levenshtein distance*** is one of the more popular edit distance functions available and Jin, Li and Mehrotra (2002) stated that it is a common measure of textual similarity. The algorithm works on the principle of assigning a unit cost to all edit operations and then determines the smallest number of insertions, deletions, and substitutions required to change one **string** into another (Grannis et al., 2004) by using dynamic programming (Winkler, 2005). A formula can be structured as follows: Metric = 1 – [LEVENSHTEIN(name1,name2) ÷ MAXLEN (NAME1, NAME2)]. A value of 1 represents an exact match, while zero indicates little similarity. To illustrate this, use the names 'TAMMY SHACKELFORD' and 'TAMMIE SHACKLEFORD' again. The Levenshtein distance value will be 4 (one substitutes 'I' for 'Y', inserts an 'E' after 'I', and reverses the order of the 'E' and 'L' (two substitutions)) and the length of the longer name is 17 (ignoring white space); thus the metric value = 1 – (4 / 17)) ≈ 0.7647

- The ***q-gram distance*** between two strings is obtained by sliding a window of length *q* over the characters of the string (Jin et al., 2002). The relative q-gram distance can be defined as:

$$\Delta_q(s_1, s_2) = 1 - \frac{|G(s_1) \cap G(s_2)|}{|G(s_1) \cup G(s_2)|}$$

For an example if s1='Harrison Ford' and s2='Harison Fort'. $G_{(s1)} \cap G_{(s2)} = 10$ and $G_{(s1)} \cup G_{(s2)} = 13$. Therefore the q-gram distance $\Delta = 1 - 10/13 \approx 0.23$. Clearly, the smaller the relative q-gram distance between two strings, the more similar they are.

- ***Monger-Elkan distance*** is a more complex yet well-tuned distance function. According to Cohen et al. *(*2003a) it is a variant of the Smith-Waterman distance function with particular cost parameters, and scaled to the interval [0,1]. Monge and Elkan propose the following *recursive matching scheme* for comparing two long strings *s* and *t* (Cohen et al., 2003a). First, *s* and *t* are broken into substrings *s* = *a*1 … *aK* and *t* = *b … bL*. Then, similarity is defined as:

$$sim(s,t) = \frac{1}{K} \sum_{i=1}^{K} \max_{j=1}^{L} sim'(A_i, B_j)$$

10.4.5. String Comparisons Methods used in Healthcare

Grannis et al. (2004) demonstrated that the ***Jaro-Winkler*** comparator achieved the highest linkage sensitivities of 97.4% and 97.7%.when they compared *raw name matching, exact phonetic name matching and* three approximate string comparators on a hospital registry dataset and a Social Security death master file (SSDMF). The approximate comparators included the modified ***Jaro-Winkler*** method, the ***longest common substring***, and the ***Levenshtein*** edit distance while the phonetic name matching included ***NYSIIS***.

Roos et al. (1986) used the Newcombe record linkage method to link administrative data where neither identifying numbers nor names were available. Gender was used to split the files and a phonetic surname code (**NYSIIS**) as the pocket identifier. Eight variables were used in the linkage process (gender, death year, death month, death day, birth year, birth month, initials and locality). Weighting was handled conservatively with disagreement on death month and death date incorporated into the weighting algorithm. 96% of individuals could be linked between the two files (Roos et al., 1986).

10.5. Chapter Summary

This chapter outlined the theory associated with record linkage. The theory will in turn provide the basis of linking data marts together in order to make a longitudinal record possible. This will be discussed in future chapters. It is important to note that the FSDOH will attempt to make use of a probabilistic record linkage system to link up all data for a patient across the individual data marts. The software system that will be used for this purpose is called GRLS. The following chapter will discuss the usage of GRLS and how this will be used to link all the individual data marts.

CHAPTER 11 - LINKING THE INDEPENDENT DATA MARTS

11.1. Introduction

In the previous chapter the theory of record linkage was discussed together with the rationale of all the key record linkage design decisions. A proposed record linkage strategy was introduced with a literature investigation on how this project will utilize probabilistic matching mechanisms in a data warehouse environment. From the literature investigation enough evidence was provided that this was not done successfully before.

This chapter will outline the plan to use probabilistic methods in linking up the relevant data fields from the following data marts and external data sources to provide a coherent longitudinal electronic patient record for all ARV patients (waiting and on treatment). The following individual data marts will be included in this plan:

- PADSDM (PADS2 clinical encounters from the inpatients database)
- ARVDM (ARV clinical encounters)
- TBDM (Tuberculosis)
- NDDM (Notifiable Diseases)
- NHLSDM (Blood results from NHLS national database)
- HOSPDM (Meditech clinical encounters from the inpatients database)

This chapter fits into the *action taking* phase of the action research cycle.

11.2. Conforming the Patient Dimension

The exercise to conform the different patient dimensions will be vital in constructing a longitudinal patient record (LPR) for all patients enrolled in the ARV treatment programme. Please note that the term LPR and the concept of longitudinal patient records will be covered in chapter 12. It is envisaged that the Patient Dimension of the ARVDM will take the center stage in this exercise and that both deterministic and probabilistic mechanisms will be deployed to construct a longitudinal patient record.

The conforming exercise is an extremely cumbersome and complicated process. The reason for this difficulty originated in the manner the data was structured in each operational data source. Each operational data source had different unique patient identifiers for identifying a patient and in some of the operational data sources a unique patient identifier was totally absent. Furthermore, the attributes for each patient was different among each operational data source. With the construction of each data mart, surrogate keys had to be created to uniquely identify each patient. In the South African context, the RSA Identify Number is often used as the unique patient identifier. See table 11-1 for a

summary of unique and valid (length must be 13 characters) patient identifiers available, across all the data marts:

Table 11-1: Summary of Unique Patient Identifiers (Information as of the 12th August 2008)

Data Mart	Table Name	Unique Person Identifier Available	Unique Person Identifier Available (%)
TBDM	TB_PATIENTS_DIM	0 / 128,086	0.00%
ARVDM	ARV_PATIENTS_DIM	47,255 / 62,947	75.07%
PADSDM	PADS2_PATIENTS_DIM	398,482 / 1,562,142	25.50%
HOSPDM	HOSP_PATIENTS_DIM	223,455 / 618,020	36.15%
NDDM	NOTIF_PATIENTS_DIM	0 / 11,445	0.00%
NHLSDM	NHLS_PATIENTS_DIM	8,908 / 94,840	9.39%

In an effort to overcome this linkage difficulty, a mapping table is proposed that will form part of the ARVDM. This mapping table can then later be turned into a **conformed patient dimension**. Record linkage mechanisms will be used in an attempt to identify the same patient across the different data marts. The patient dimension tables will then be compared with each other to populate this **mapping table.** Depicted below (see figure 11-1) is a conceptual view of the proposed mapping process with the mapping table taking centre stage.

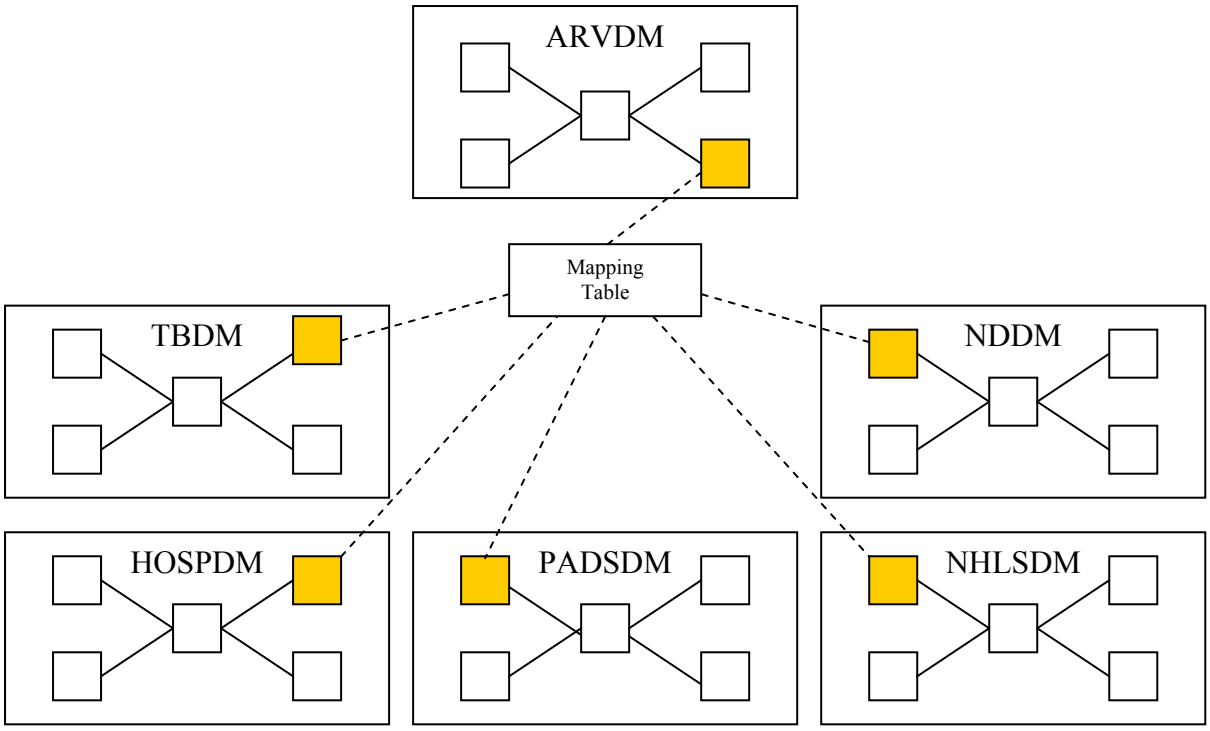


Figure 11-1: Proposed Mapping Process (Patient Dimension)

Surrogate keys will then be created for each patient from the different dimension tables. See table 11-2 as an example of the proposed mapping table.

Table 11-2: Proposed Mapping Table

Global Patient Identifier	ARVDM Surrogate Key (Patient Dim)	Patient Name	TBDM Surrogate Key (Patient Dim)	Notifiable Surrogate Key (Patient Dim)	PADS Key (Patient Dim)	NHLS Key (Patient Dim)
PK=12	PK=1002	JP Doe	PK=45309	PK=10932	-	-

The second part of this process will be to construct the longitudinal patient record by linking all the different facts together in a single data mart and then turning this mapping table into a conformed patient dimension for this data mart. See figure 11-2 for the proposed dimensional model. In theory the end result will be a longitudinal patient record, but now in a data warehousing environment. The effectiveness of this approach will be tested, but it is anticipated that a patient could now be traced across all the data marts and provide a single electronic patient record, all linked together by the mapping table.

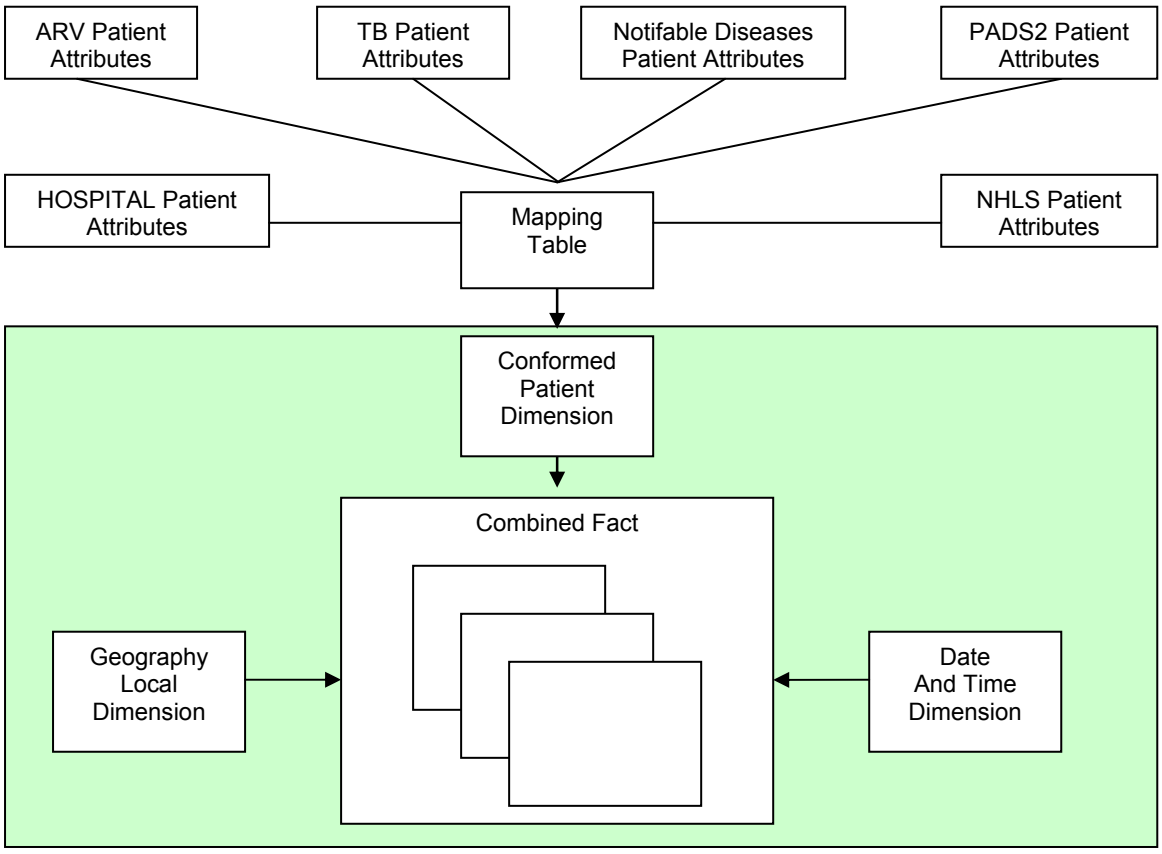


Figure 11-2: Proposed Combined Data Mart

For the construction of a mapping table, a modified ETL process is suggested that will include the de-duplication and mapping of data, once loaded. In theory this additional process will ***lengthen*** the entire ETL process and could add more time to it than was previously observed. The suggested steps can be structured and formalized as follows:

- Extraction
- Transformation
- Loading
- **Linkage**

11.3. Rationale of using probabilistic record linkage

By thoroughly examining all the different patient dimension tables across the different data marts, the following observations were made to strengthen the rationale of using probabilistic record linkage methods:

- Lack of an unique identifier such as the South African Identification Number in the patient dimension of the TBDM and NDTM.
- Where the unique identifier was present, it was poorly populated (for example 63% of the records in the PADSDM contained no value). Refer to section 11.2 for a detailed layout of the absence of a unique identifier.
- Different spellings of a SURNAME, FIRSTNAME for the same patient.
- Different combination of INITIALS for the same patient.
- Missing values in GENDER and BIRTHDATE, or BIRTHDATE that will differ because of the longitudinal nature of records in the different data marts.

11.4. Probabilistic Record Linkage and GRLS

The goal of record linkage is to determine which records refer to the same entity. Probabilistic record linkage uses *probabilities* and *record pairs* to quantify the *likelihood* of a pair being correctly linked (refers to the same entity).

According to the GRLS Concept Guide (Statistics Canada, 2007:3), the primary objective of **GRLS** is to classify the record pairs into two sets: a **linked** set (**L**) and a **nonlinked** set (**N**). The following paragraphs will provide the background knowledge on the workings of GRLS and is extracted from the GRLS Concept Guide.

11.4.1. Rule Outcomes

When GRLS compare pairs, **outcomes** are produced. These rule outcomes can be classified into: A (Full Agreement), P (Partial Agreement), D (Disagreement) and M (Missing). Full agreement is when the fields are the same, partial agreement is when fields are similar, disagreement is when fields are not the same and are not similar and missing is when one or both fields are blank.

11.4.2. Odds Ratios

With rule outcomes defined, the next step is to determine whether a pair of records is likely to be a **linked** set (L) or a true **non-link** and hence a member of the **non-linked** set (N). In order to achieve this a measurement is needed of how likely it is that a particular **rule outcome** vector **R** occurs for pairs in set **L** versus how likely it is that **R** occurs for pairs in set **N**. An **odds ratio** is used to specify this measurement and can be determined and stated in terms of the ratio of conditional probability as follows:

$$OR = \frac{P(R \mid L)}{P(R \mid N)}$$

Each rule is assigned an odds ratio and the sum of all odds ratios provides the outcome probabilities for a linked set. Depicted below (see figure 11-3) is an example using rules on SURNAME, BIRTHYR and SEX. For this example the Surname rule linked set sum is 1.00 (.70 + .20 + .05 + .05)

Outcome Probabilities

Rules:	SURNAME				BIRTHYR				SEX		
	A	PA	M	D	A	PA	M	D	A	M	D
Linked Set	.70	.20	.05	.05	.80	.10	.05	.05	.95	.02	.03
Nonlinked Set	.01	.04	.05	.90	.02	.05	.05	.88	.49	.02	.49
Odds Ratios	70.00	5.00	1.00	.06	40.00	2.00	1.00	.06	1.94	1.00	.06

Figure 11-3: Example using rules on SURNAME, BIRTHYR and SEX

These probabilities are stating that for pairs of records belonging to the **linked** set, the Surname rule has an outcome of:

- A (agreement) 70% of the time
- PA (partial agreement) 20% of the time
- M (missing) 5% of the time
- D (disagreement) 5% of the time

11.4.3. Frequency Probabilities

All agreement (**A**) and some partial agreement (**PA**) rule **outcomes** will have an associated **result** equal to the value in agreement, or to the type of agreement. In those cases a Frequency Probability for the specific value can be used in place of the Outcome Probability in the calculation of the Odds Ratio. If both linked and non-linked frequency probabilities are used, then a particular result would be the ratio of the conditional probability as shown below:

$$OR = \frac{P(R_i \mid L)}{P(R_i \mid N)}$$

11.5. GRLS Workings

GRLS record linkage can be divided into *internal linkage* or *two-file linkage*. An internal linkage is performed on a single file that contains more than one record for a given entity. This is often performed to identify duplicate records and to then group the records together by entity. Two-file linkage is performed on two files that can contain information for the same entities. For the purpose of this study both internal and two-file linkage methods will be used. Internal linkage will be employed on each dataset to identify duplicate patients and to group them together. After all the identified data marts were subjected to internal linkage, each internal linked file (of the relevant data mart) will then be matched to the ARV data mart using two-file linkage. Based on statistical decision theory, GRLS breaks the linkage operation into three major phases (Search, Decide and Group) as depicted in figure 11-4.

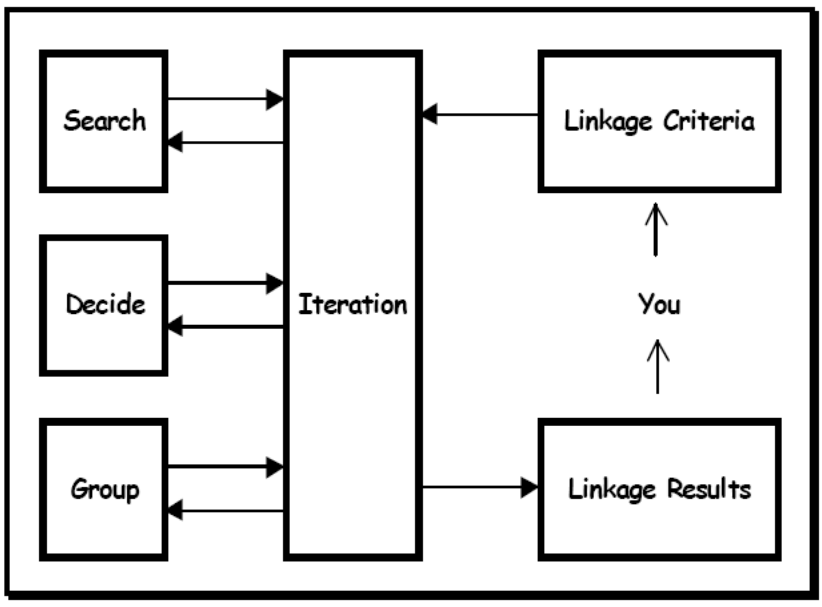


Figure 11-4: Three Major Phases of GRLS (Statistics Canada, 2007:12)

These major phases can be applied to both internal linkage and two-file linkage. The three major phases will now be briefly described.

11.5.1. Search Stage

A searching phase is implemented where a table of all the potential pairs is generated using the initial criteria set by the user. This is often called the first set of ***potential pairs***. It is important to note that this phase can take an extremely long time. For example, define *n* as the row counts of each table. Due to the Cartesian product aspect (Sauleau et al., 2005) at least $(n^2-n)/2$ comparisons will be required if one compare each record to every other record. Blocking is often used in this phase to minimize the number of potential pairs. GRLS uses deterministic methods and is totally user-defined in this phase.

The following actions are performed in GRLS in this stage and can be summarized as follows:

- Specify the files (loaded into ORACLE database tables) to be linked
- Specify match conditions for creating potential pairs
- Specify rules for comparing the input records
- Specify outcome probabilities to be assigned to rule outcomes

11.5.2. Decision Stage

A decision phase is carried out where linkage rules are applied to the potential pairs and weights or odds ratios are assigned and the potential pairs are divided into sets of definite (Def), possible (Pos) and excluded (Excl) pairs by the setting of upper (TU) and lower (TL) thresholds. Each linkage rule produces an outcome of missing, agreement, partial agreement, or disagreement. It is possible to refine the linkage weights and reset the threshold values using the possible and definite pairs that are then classified as definite, possible or rejected (Rej) as depicted in figure 11-5.

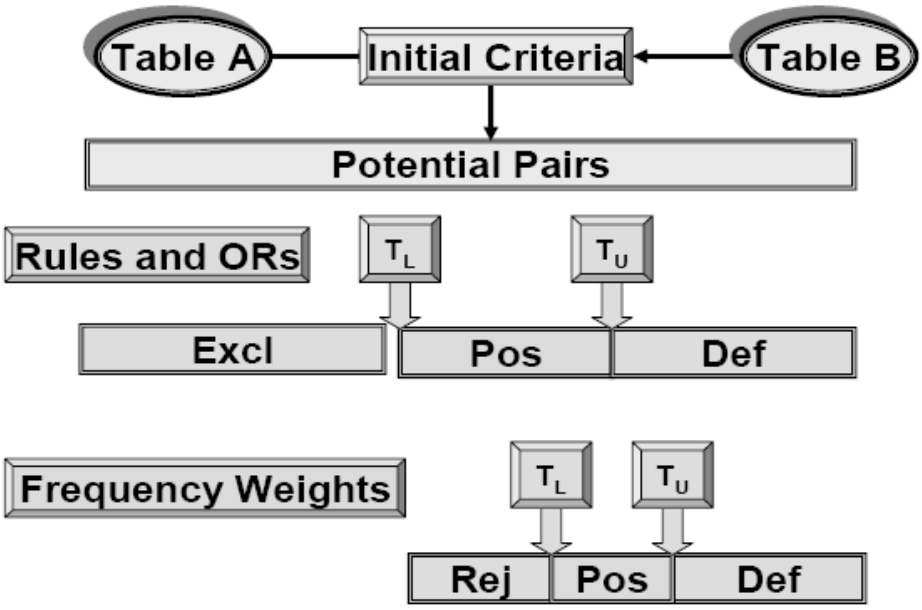


Figure 11-5: Implementation of linkage operation (Fair, 2004:44)

The following actions are performed in GRLS in this stage and can be summarized as follows:

- Adjust the odds ratio for existing linked record pairs without re-comparing the input data.
- Alter the thresholds. These actions will change the distribution of the record pairs, for example record pairs that were linked are now non-linked and vice versa.
- Revise outcome probabilities and automatically *apply* them to the record pairs in this stage.
- Calculate and apply frequency probabilities which are based on the field values in the records.

After adjusting the **odds ratios** for the rule outcomes and frequency probabilities, the pairs must be re-classified. Pairs with adjusted **odds ratios** below the *lower* **threshold** are classified as *rejected* and are considered to be in set **N** (**non-linked** set). All possible record pairs can be divided into two populations: those record pairs which are *truly linked*, and those which are *truly non-linked*. The primary goal of the linkage project is to find the members of the *truly linked* population. Because it represents all possible record pairs which do not link, the true non-linked population will be far greater

than the true linked one. A problem exists with the overlap between the two populations (the “green” area) because it is not obvious to which set the record pair belongs. If the data is complete and accurate then the “green” area can be minimized, forcing a better separation between the **non-linked** and **linked** pairs. Depicted below (figure 11-6) is an example of the distribution of record pairs.

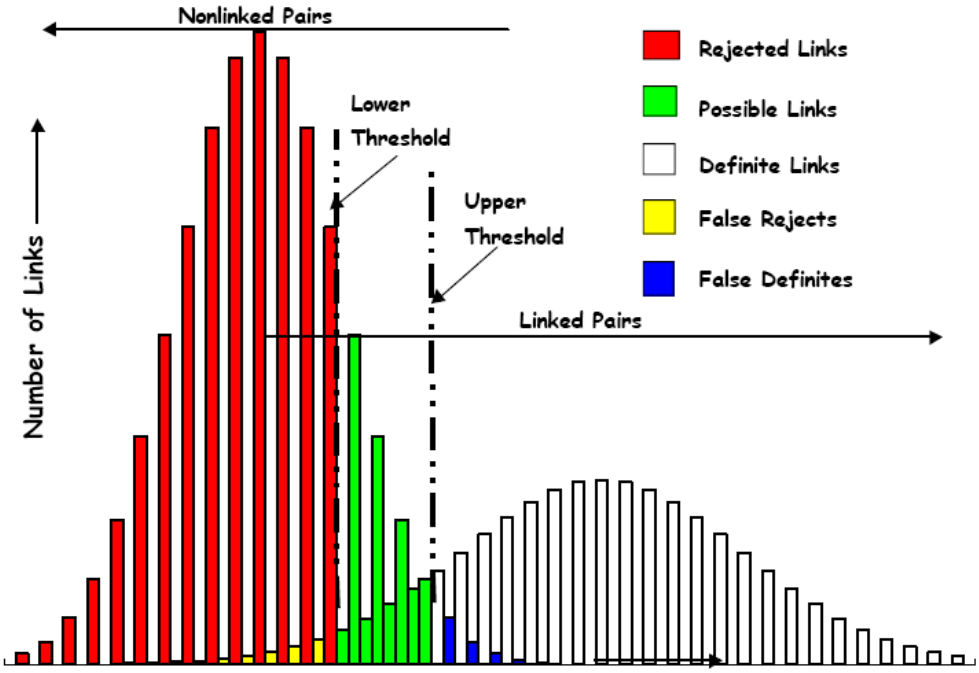


Figure 11-6: Pair Odds Ratio (Statistics Canada, 2007:13)

The next step is to examine all or selected pairs and then reset the thresholds to change the number of pairs in the set L, possibly linked (“green” area) and set N. GRLS allows the user to manually or automatically assign pairs in the “green” area to either set L or set N. The outcome of the search stage is to minimize possible pairs while at the same time ensuring rejected links and definite links pairs are close to being correct.

11.5.3. Grouping Stage

In the grouping phase, the pairs of possible and definite links which could relate to the same entity are grouped together in a manner specified by the user, and the final groups are generated. Manual resolution may also be carried out during the process and the results updated.

Records are grouped according to the status of the links between them as is illustrated in figure 11-7. Records may have just one link to another record, or they may have links to several records. Records joined either by **possible** or **definite** links are arranged into **weak groups** which can be quite large. Within **weak groups**, records joined by **definite** links *only* are further arranged into **strong groups**. A **weak group** may therefore contain several **strong groups**, and it is the **strong groups** that contain the best links.

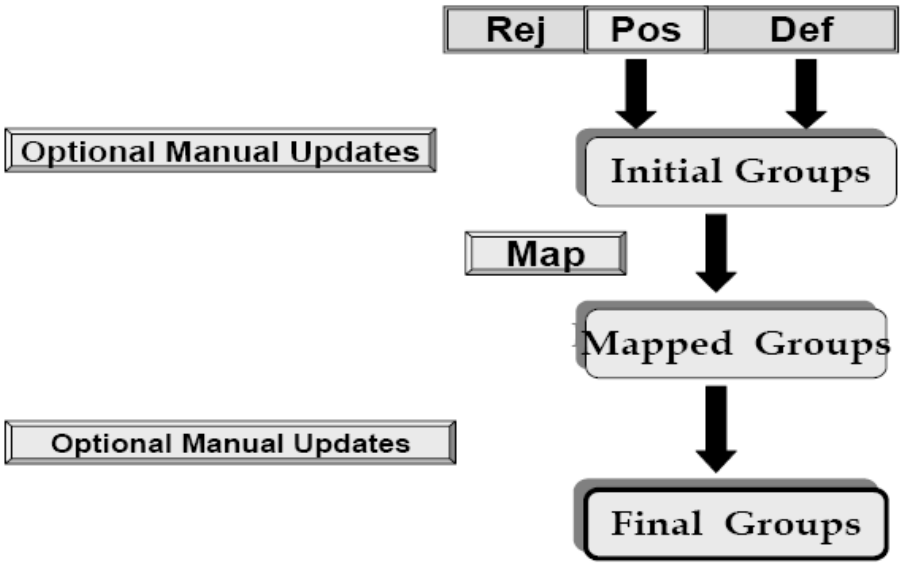


Figure 11-7: Grouping of pairs of records (Fair, 2004:45)

The following actions are performed in GRLS in this stage and can be summarized as follows:

- GRLS can resolve conflicts automatically. This is called mapping. In this case linkage requirements are specified, for example one-to-one, many-to-one and or one-to-many.
- GRLS also provides the functionality to resolve the conflicts manually and this is done by on-screen updating of group contents. This is sometimes referred to as manual resolution of group conflicts.

11.5.4. Environment and Workflow Process

The GRLS server used for the probabilistic linkage experiments was situated at the University of Bern in Switzerland. The researcher used the Secure Windows Remote Desktop application to access the GRLS software. The GRLS server environment was deployed on a Red Hat Linux server with an Oracle 10 Release 2 database instance. Depicted below (figure 11-8) is the workflow process involved in a GRLS probabilistic linkage exercise. This is applicable to both internal and two-file linkage.

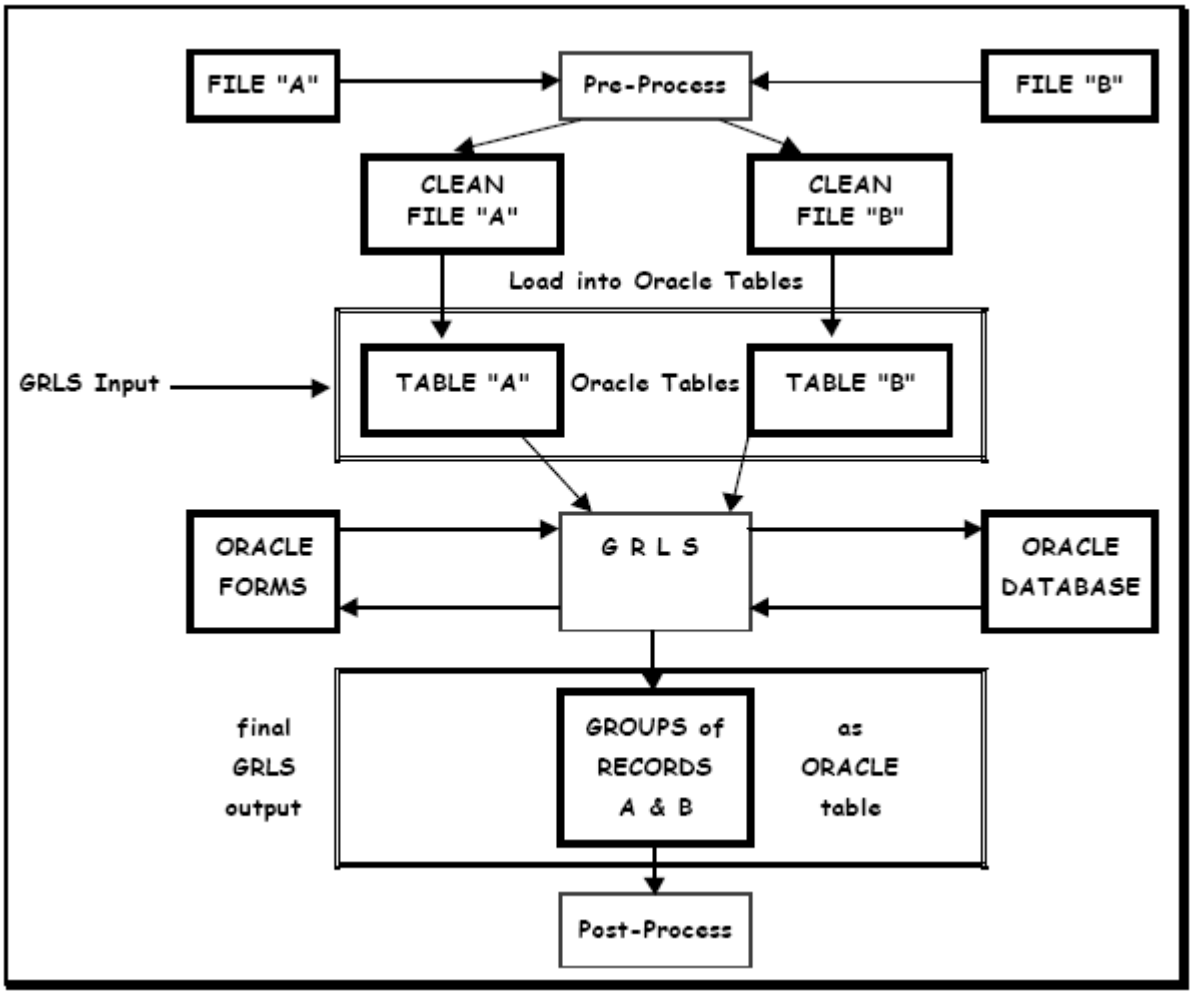


Figure 11-8: Environment of a GLRS probabilistic linkage experiment (Statistics Canada, 2007:15)

The two files, referred to as File **A** and File **B**, must be pre-processed before loading them into **ORACLE** tables. This pre-processing, which is done outside of **GRLS**, includes performing a variety of tasks. Some of these tasks include:

- Creating unique record identifiers (**recid**)
- Identifying duplicate records using a **dupflag** field
- Standardizing name and addresses (e.g., drop unwanted titles such as Mr., Mrs., Jr., etc.)
- Creation of additional data columns which will be used in match conditions for creating **potential pairs** (e.g. use **NYSIIS** to phonetically encode a surname field).

11.6. Internal Record Linkage

The following section will describe the internal record linkage that was performed on the different data marts and the lessons learned.

11.6.1. Notifiable Diseases (NTDM)

Both deterministic linkage and probabilistic linkage were performed on the patient dimension table (NOTIF_PATIENT_DIM) in the NTDM data mart. At the time of conducting this linkage experiment the NOTIF_PATIENT_DIM table contained 11,445 entries.

11.6.1.1. Deterministic Linkage

The deterministic linkage outcomes using Oracle SQL joins on different column combinations are summarized in table 11-3. The calculated field BIRTHYEAR was a calculation between NOTIFIED_DATE and the AGE and added for linkage purposes. All BIRTHYEAR values were set to the Year + 01/01 as no other information on month or day was available.

Table 11-3: Deterministic Linkage Outcomes for NOTIFDM

ORACLE SQL JOINS	Set 1
	11,445
• Records with ID Number and length=13	0
• Duplicate records : Using Birth Year, Gender, Surname and Initials	1,908
• Duplicate records : Using Birth Year, Gender, Surname and First name	1,383
• Duplicate records: Using Birth Year, Gender, Surname, First name and Initials	1,379
• Duplicate records: Using ID number	0

11.6.1.2. Probabilistic Linkage

Before GRLS linkage can be performed a table with all the fields needed for matching must be created. A table called PROJARV_NOTI_001 was created by extracting fields from the NOTIF_PATIENTS_DIM. Additional columns were added such as RECID and DUPFLAG which is a requirement of GRLS. Depicted in figure 11-9 is the table definition of PROJARV_NOTI_001.

COLUMN ID	Column Name	Data Type	Nullable	Data Default	Primary Key
1	RECID	NUMBER(8,0)	No	(null)	1
3	DUPFLAG	NUMBER(2,0)	No	(null)	2
2	PATID	VARCHAR2(5 BYTE)	Yes	(null)	(null)
4	SURNAME	VARCHAR2(29 BYTE)	Yes	(null)	(null)
5	FIRSTNAME	VARCHAR2(29 BYTE)	Yes	(null)	(null)
6	GENDER	VARCHAR2(1 BYTE)	Yes	(null)	(null)
7	RACE	VARCHAR2(8 BYTE)	Yes	(null)	(null)
8	P_TOWN	VARCHAR2(18 BYTE)	Yes	(null)	(null)
9	P_DISTRICT	VARCHAR2(4 BYTE)	Yes	(null)	(null)
10	P_SUBDISTRICT	VARCHAR2(5 BYTE)	Yes	(null)	(null)
11	BIRTHDATE	VARCHAR2(8 BYTE)	Yes	(null)	(null)
12	AGE	VARCHAR2(2 BYTE)	Yes	(null)	(null)
13	FACILITY	VARCHAR2(4 BYTE)	Yes	(null)	(null)
14	F_TOWN	VARCHAR2(14 BYTE)	Yes	(null)	(null)
15	F_DISTRICT	VARCHAR2(4 BYTE)	Yes	(null)	(null)
16	F_SUBDISTRICT	VARCHAR2(5 BYTE)	Yes	(null)	(null)
17	DISEASE_NAME	VARCHAR2(37 BYTE)	Yes	(null)	(null)
18	P_ADDRESS	VARCHAR2(54 BYTE)	Yes	(null)	(null)
19	P_ADDRESS_NR	VARCHAR2(9 BYTE)	Yes	(null)	(null)
20	NOTI_DATE	VARCHAR2(8 BYTE)	Yes	(null)	(null)

Figure 11-9: Table definition of PROJARV_NOTI_001

Rules and Weights

For probabilistic matching to be performed, rules must be defined and created in GRLS. Table 11-4 outlines the rules and a short description of each rule that was created.

Table 11-4: List of rules used in GRLS for NOTIFDM

Rule	Rule #	Rule Description (to review pairs only with)
F_FACILITY	10	Comparison on the facility name
F_TOWN	20	Comparison on the town name of the facility
F_S_DIST	30	Comparison on the sub district name of the facility
F_DIST	40	Comparison on the district name of the facility

Rule	Rule #	Rule Description (to match pairs with)
P_TOWN	50	Comparison on the town name of the patient
P_S_DIST	60	Comparison on the sub district name of the patient
P_DIST	70	Comparison on the district name of the patient
GENDER	80	Comparison on the gender
RACE	90	Comparison on the race (not used, instead RACE_2 was used)
BIRTHYEAR	100	Comparison on the year of birth

Rule	Rule #	Rule Description (to match pairs with)
FSTNAME	110	Comparison on first name
SURNAME	120	Comparison on surname
FS_NAME	130	Comparison on surname = first name and first name = surname
X_NAME	140	Comparison on first name and surname in case they are switched. Only execute this rule if SURNAME != A, and FSTNAME != A and FS_NAME != A
DISEASE	150	Comparison on the notifiable disease
ADRESS	160	Comparison on the street name
ADRESSNR	170	Comparison on the street number
RACE_2	180	Additional comparison on race, with additional code added to treat AFRICAN and COLOURED the same.

With the process of rule creation completed, weights had to be assigned to each of the possible rule outcomes of D (DISAGREE), M (MISSING) and A (AGREE). Table 11-5 outlines the rules with their respective weights for each rule outcome.

Table 11-5: Rules with their respective weights and probabilities in GRLS for NOTIFDM

rule	rule#	outcome	algorithm	precision	weight linked	weight non-linked
F_FACILITY	10	D			0	0
F_FACILITY	10	M			0	0
F_FACILITY	10	A	C1[1]	4	0	0
F_TOWN	20	D			0	0
F_TOWN	20	M			0	0
F_TOWN	20	A	C1	14	0	0
F_S_DIST	30	D			0	0
F_S_DIST	30	M			0	0
F_S_DIST	30	A	C1	5	0	0
F_DIST	40	D			0	0
F_DIST	40	M			0	0
F_DIST	40	A	C1	4	0	0
P_TOWN	50	D			-20	2
P_TOWN	50	M			0	0
P_TOWN	50	A	C1	18	-4	37
P_S_DIST	60	D			-25	3
P_S_DIST	60	M			0	0
P_S_DIST	60	A	C1	5	-3	33

[1] left-to-right character comparison (number of characters compared equal to the precision)

rule	rule#	outcome	algorithm	precision	weight linked	weight non-linked
P_DIST	70	D			-33	4
P_DIST	70	M			0	0
P_DIST	70	A	C1	4	-2	23
GENDER	80	D			-66	10
GENDER	80	M			0	0
GENDER	80	A	C1	1	0	10
RACE	90	D			0	0
RACE	90	M			0	0
RACE	90	A	C1	8	0	0
BIRTHYEAR	100	D			-60	0
BIRTHYEAR	100	M			0	0
BIRTHYEAR	100	A	Y1[2]	1	-5	20
BIRTHYEAR	100	A	Y2	2	-15	0
BIRTHYEAR	100	A	Y3	3	-30	0
FSTNAME	110	D			-35	5
FSTNAME	110	M			0	0
FSTNAME	110	A	C1	29	-10	40
FSTNAME	110	A	S1[3]	29	-8	25
FSTNAME	110	A	T1[4]	0	0	10
FSTNAME	110	A	X1[5]	90	0	15
FSTNAME	110	A	P1[6]	5	0	5
SURNAME	120	D			-55	5
SURNAME	120	M			0	0
SURNAME	120	A	C1	29	-20	50
SURNAME	120	A	S1	29	-8	30
SURNAME	120	A	T1	0	0	10
SURNAME	120	A	X1	90	0	15
SURNAME	120	A	P1	5	0	5
FS_NAME	130	D			-18	15
FS_NAME	130	M			0	0
FS_NAME	130	A	U1[7]	16	-10	45

[2] difference in years (less than or equal to the precision)

[3] phonetic matches using **soundex.**

[4] typo matching (exact match, transposition of one character, mismatch of one character, extra character)

[5] string proximity - This function returns value from 0 to 100 where 100 is a perfect match. The **outcome** is satisfied if value returned is >= the **precision.**

[6] complete string on one table matches the last characters of the string on the other table

[7] user defined rule that was coded in C

rule	rule#	outcome	algorithm	precision	weight linked	weight non-linked
X_NAME	140	D			0	0
X_NAME	140	M			0	0
X_NAME	140	A	U1	16	-10	40
DISEASE	150	D			0	0
DISEASE	150	M			0	0
DISEASE	150	A	C1	37	0	0
ADRESS	160	D			-5	4
ADRESS	160	M			0	0
ADRESS	160	A	C1	54	-2	27
ADRESS	160	A	S1	54	-5	13
ADRESS	160	A	X1	92	-3	16
ADRESS	160	A	P1	5	-7	11
ARDESSNR	170	D			-5	4
ARDESSNR	170	M			0	0
ARDESSNR	170	A	C1	9	-2	12
RACE_2	180	D			-48	4
RACE_2	180	M			0	0
RACE_2	180	A	U1	16	-1	20
RACE_2	180	A	U1	16	-2	20

Search Stage Actions

In this stage a set of **potential pairs** is created by specifying a match condition using some of the table fields. At the time the internal linkage was performed, the **GRLS NDDM Patient Table** contained 11,445 record entries. Two synonyms were created called XA and XB, both referring to the same table. By using the Cartesian product between XA and XB the initial number of pairs was calculated as 130,988,025 record pairs. Using blocking and deterministic methods this number was reduced to only 15,857,510 record pairs. The initial threshold was set to (-75, 25, 100) to allow for the maximum number of possible pairs. Depicted below (see figure 11-10) is the create pair selection criteria that was used to reduce the initial number of potential pairs.

```
(
  XA.FIRSTNAME = XB.FIRSTNAME
  OR
  XA.SURNAME = XB.SURNAME
  OR
  ( XA.GENDER = XB.GENDER
    AND
    XA.F_DISTRICT = XB.F_DISTRICT
  ) OR
  ( XA.BIRTHDATE = XB.BIRTHDATE
    AND
    XA.F_DISTRICT = XB.F_DISTRICT
  ) OR
  ( XA.GENDER = XB.GENDER
    AND
    XA.P_DISTRICT = XB.P_DISTRICT
  ) OR
  ( XA.BIRTHDATE = XB.BIRTHDATE
    AND
    XA.P_DISTRICT = XB.P_DISTRICT
  )
)
```

Figure 11-10: Selection criteria to create the initial record pairs

Decide Stage Actions

The initial pair creation process produced 2,999 potential duplicate pairs. Adjustments to the odds ratios and thresholds had to be made on an iterative approach. The primary outcome of this stage is to find members truly linked to each other.

A frequency probability was introduced on GENDER to provide better linkage results and reduce the number of possible links. BLACK and COLOURED was given a weight of 5, WHITE a weight of 35 and ASIAN a weight of 40. These frequency weights were manually set to increase the accuracy of the linkage algorithms. BLACK and COLOURED was given the same weight of 5. This was done to nullify a data capturing error where in many instances the race was wrongly identified and captured. For example COLOURED was wrongly selected for a black person, and similarly BLACK was wrongly selected for a coloured person.

The original threshold of (-75, 25,100) was adjusted until the number of possible and definitive links provided acceptable results. The final threshold was (65,105), which will be used in the Group Stage Actions.

Group Stage Actions

Using the final threshold of (65,105), the 2,999 pairs were divided into three groupings. Group 1 consisted of 1,437 DEFINITE matched pairs, group 2 consisted of 657 POSSIBLE matched pairs and group 3 consisted of 905 REJECTED matched pairs. The next step was to map the 1,437 DEFINITE matches (above the 105 total weight threshold) into strong and weak groups using GRLS automatic one-to-one mapping. A total of 1,107 unique strong groups were identified in this step. These strong

groups produced a final data set of 2,402 duplicate patient entries out of the 11,445 entries which had a total weight of above 105.

Converting the Mapped Link Table

With the mapping process completed, the final product is a LINK table stored within the GRLS database. This table contains all the one-to-one mappings, weak groups, strong groups and the total weight. This table is exported and then imported into the FSDOH data warehouse and renamed to a name associated with the data mart. For example LINK was renamed to NOTIF_LINK for this data mart and it is depicted below (see figure 11-11).

COLUMN ID	Column Name	Data Type
1	LINKID	NUMBER
2	RECID_1	NUMBER
3	DUPFLAG_1	NUMBER(2,0)
4	RECID_2	NUMBER
5	DUPFLAG_2	NUMBER(2,0)
6	ENSEMBLE	CHAR(1 BYTE)
7	STATUS	CHAR(1 BYTE)
8	SOURCE	CHAR(1 BYTE)
9	TOTWGHT	NUMBER
10	WEAKGRP	NUMBER
11	STRONGGRP	NUMBER
12	LINKSET	NUMBER

Figure 11-11: NOTIF_LINK table

The final step in the internal linkage exercise is to convert the NOTIF_LINK table into a table associated with the original NOTIF_PATIENTS_DIM table and the PROJARV_NOTI_001 table used by GRLS. SQL code was developed to perform this conversion task. The primary focus was to merge the NOTIF_LINK table with the PROJARV_NOTI_001 table. A new temporary table called GRLS_NOTIF_PATIENTS_LINKED was created to help with this merging. All the records for the RECID_1 and RECID_2 combination were inserted together with the LINKID and the TOTWGHT. One needs to remember that this combination must be reversed to provide a merge on RECID_2 as well. Therefore, the RECID_2 and RECID_1 combination was inserted again. This step resulted in duplicate entries. By comparing each individual TOTWGHT to the highest TOTWGHT in a strong group and weak group, a single data row would be obtained for each strong and weak group combination.

For example by using the LINKID=3709 from the NOTIF_LINK table (see table 11-6), the newly converted rows will now be available in GRLS_NOTIF_PATIENTS_LINKED (see table 11-7) for future use.

Table 11-6: Example from the NOTIF_LINK table

LINKID	RECID_1	RECID_2	TOTWGHT	WEAKGRP	STRONGGRP
3709	43290	48653	147	325	1

Table 11-7: Example from the GRLS_NOTIF_PATIENTS_LINKED table

NewID	LinkID	RecID	Weight	Weak Grp	Strong Grp	Dupl	Surname	Firstname
4201	3709	**43290**	147	42	1	1	CEBE	GOLDEN
4201	3709	**48653**	147	42	1	1	CEBE	GOLDEN

11.6.1.3. Linkage Findings Discussion

The most accurate multicolumn SQL join produced 1,378 duplicates entries. This represents a deterministic matching result on the following columns: Birth Year, Gender, Surname, First name and Initials. GRLS produced 1,437 DEFINITES pairs using probabilistic matching. When converting the DEFINITE pairs into individual row entries, GRLS identified 2,402 duplicate entries.

Using the newly identified duplicate entries (2,402) the dimension table had to be de-duplicated. The GRLS_NOTIF_PATIENTS_LINKED table containing 11,445 table rows was reduced to 10,164 table rows. This is a direct reduction of **1,281** table rows (**11.19%**). This de-duplicated table will be used for the two-file linkage which will be discussed later in the chapter.

11.6.2. NHLS Blood Results (NHLSDM)

Both deterministic linkage and probabilistic linkage were performed on the patient dimension table (NHLS_PATIENTS_DIM) in the NHLS data mart. At the time of conducting this linkage experiment the NHLS_PATIENTS_DIM table contained 94,810 entries.

11.6.2.1. Deterministic Linkage

The NHLS_PATIENTS_DIM table was logically partitioned into five sets of groups to enable GRLS to perform the record linkage. The same five set groupings where used for both deterministic linkage and probabilistic linkage.

The deterministic linkage outcomes using Oracle SQL joins on different column combinations are summarized in table 11-8.

Table 11-8: Deterministic Linkage Outcomes for NHLSDM

ORACLE SQL JOINS	Set 1	Set 2	Set 3	Set 4	Set 5	Total
	20,000	20,000	20,000	20,000	14,810	**94,810**
• Records with ID Number and length=13	2,386	2,177	2,198	2,384	1,766	**10,911**
• Duplicate records : Using Birth Date, Gender, Surname and First name	11,396	11,393	11,242	11,368	8,450	**53,849**
• Duplicate records : Using Birth Date, Gender, Surname and Initials	11,697	11,674	11,589	11,657	8,703	**55,320**
• Duplicate records: Using Birth Date, Gender, Surname, First name and Initials	11,396	11,393	11,242	11,368	8,450	**53,849**
• Duplicate records: Using Birth Date, Gender, Surname, First name, Initials and ID Number	1,709	1,577	1,596	1,717	1,312	**7,911**
• Duplicate records: Using ID number	1,758	1,605	1,634	1,757	1,350	**8,104**

11.6.2.2. Probabilistic Linkage

Rules and Weights

Table 11-9 outlines the rules and a short description of each rule that was created in GRLS for the probabilistic record matching.

Table 11-9: List of rules used in GRLS for NHLSDM

Rule	Rule #	Rule Description
GENDER	80	Comparison on the gender
BDATE	100	Comparison on the date of birth
FSTNAME	110	Comparison on first name
SURNAME	120	Comparison on surname
FS_NAME	130	Comparison on surname = first name and first name = surname
X_NAME	140	Comparison on first name and surname in case they are switched. Only execute this rule if SURNAME != A, and FSTNAME != A and FS_NAME != A
ID_NUMBER	150	Comparison on the RSA Identification Number
TESTLOC	160	Comparison on the test location

Decide Stage and Group Stage Actions

With the initial pairs of each data set available from the Search Stage, adjustments to the odds ratios and thresholds had to be made on an iterative manner. The original threshold of (-75, 25, 100) was adjusted until the number of possible and definitive links provided acceptable results. Tabulated

below (see table 11-10) are the final outcomes using thresholds that was providing the maximum number of true DEFINITES.

Table 11-10: Outcomes of Decide and Group Stage for NHLSDM

GRLS Outcomes	Set 1	Set 2	Set 3	Set 4	Set 5	Total
	20,000	20,000	20,000	20,000	14,810	**94,810**
Final Threshold	(40,56)	(40,60)	(40,60)	(40,56)	(40,56)	**-**
DEFINITE	11,872	11,632	11,625	11,909	8,933	**55,971**
POSSIBLE	1,440	7,610	16,192	2,470	2,173	**29,885**
REJECTED	721	1,620	2,559	901	729	**6,530**
Total Pairs	**14,033**	**20,862**	**30,376**	**15,280**	**11,835**	**92,386**

11.6.2.3. Linkage Findings Discussion

The most accurate multicolumn SQL join produced 53,849 duplicates. This represents a deterministic matching result on the following columns: Birth Date, Gender, Surname, First name and Initials. GRLS produced 55,971 DEFINITES pairs using probabilistic matching. When converting the DEFINITES pairs into individual row entries, GRLS identified 57,810 duplicate entries.

Using the newly identified duplicate entries (57,810) the dimension table had to be de-duplicated. The GRLS_NHLS_PATIENTS_LINKED table containing 94,810 table rows was reduced to 58,378 table rows. This is a direct reduction of **36,432** table rows (**38.42%**). This de-duplicated table will be used for the two-file linkage which will be discussed later in the chapter.

11.6.3. ARVDM

Both deterministic linkage and probabilistic linkage were performed on the patient dimension table (ARV_PATIENTS_DIM) in the ARV data mart. At the time of conducting this linkage experiment the ARV_PATIENTS_DIM table contained 65,300 entries.

11.6.3.1. Deterministic Linkage

The ARV_PATIENTS_DIM table was logically partitioned into two sets of groups to enable GRLS to perform the record linkage. The same two set groupings where used for both deterministic linkage and probabilistic linkage.

The deterministic linkage outcomes using Oracle SQL joins on different column combinations are summarized in table 11-11.

Table 11-11: Deterministic Linkage Outcomes for ARVDM

ORACLE SQL JOINS	Set 1	Set 2	Total
	30,000	35,300	**65,300**
• Records with ID Number and length=13	22,838	26,857	**49,695**
• Duplicate records : Using Birth Date, Gender, Surname and First name	312	467	**779**
• Duplicate records : Using Birth Date, Gender, Surname and Initials	1,143	1,615	**2,758**
• Duplicate records: Using Birth Date, Gender, Surname, First name and Initials	292	437	**729**
• Duplicate records: Using Birth Date, Gender, Surname, First name, Initials and ID Number	22	36	**58**
• Duplicate records: Using ID number	664	755	**1,419**

11.6.3.2. Probabilistic Linkage

Rules and Weights

Table 11-12 outlines the rules and a short description of each rule that was created in GRLS for the probabilistic record matching.

Table 11-12: List of rules used in GRLS for ARVDM

Rule	Rule #	Rule Description
P_TOWN	50	Comparison on the town name of the patient
P_S_DIST	60	Comparison on the sub district name of the patient
P_DIST	70	Comparison on the district name of the patient
GENDER	80	Comparison on the gender
BIRTHDATE	100	Comparison on the date of birth
FSTNAME	110	Comparison on first name
SURNAME	120	Comparison on surname
FS_NAME	130	Comparison on surname = first name and first name = surname
X_NAME	140	Comparison on first name and surname in case they are switched. Only execute this rule if SURNAME != A, and FSTNAME != A and FS_NAME != A
ADDRESS	150	Comparison on the street name
ADDRESS_NR	160	Comparison on the street number
MARRIED	170	Comparison on marital status
ID_NUMBER	180	Comparison on the ID Number (including allowing a disagree on 1 numeric value)

Decide Stage and Group Stage Actions

With the initial pairs of each data set available from the Search Stage, adjustments to the odds ratios and thresholds had to be made on an iterative approach. The original threshold of (-75, 25, 100) was adjusted until the number of possible and definitive links provided acceptable results.

Tabulated below (see table 11-13) are the final outcomes using thresholds that was providing the maximum number of true DEFINITES.

Table 11-13: Outcomes of Decide and Group Stage for ARVDM

GRLS Outcomes	Set 1	Set 2	Total
	33,000	35,300	**65,300**
Final Threshold	(50,93)	(50,95)	**-**
DEFINITE	604	793	**1,397**
POSSIBLE	579	1,182	**1,761**
REJECTED	708	14,768	**15,476**
Total Pairs	**1,891**	**16,743**	**18,634**

11.6.3.3. Linkage Findings Discussion

The most accurate multicolumn SQL join produced 729 duplicates entries. This represents a deterministic matching result on the following columns: Birth Date, Gender, Surname, First name and Initials. GRLS produced 1,397 DEFINITES pairs using probabilistic matching. When converting the DEFINITES pairs into individual row entries, GRLS identified 2,625 duplicate entries.

Using the newly identified duplicate entries (2,625) the dimension table had to be de-duplicated. The GRLS_ARV_PATIENTS_LINKED table containing 65,300 table rows was reduced to 63,944 table rows. This is a direct reduction of **1,356** table rows (**2.07%**). This de-duplicated table will be used for the two-file linkage which will be discussed later in the chapter.

11.6.4. PADSDM

Both deterministic linkage and probabilistic linkage were performed on the patient dimension table (PADS2_PATIENTS_DIM) in the PADS2 data mart. At the time of conducting this linkage experiment only data of 2008 were used. The selection criteria used were that the admission was after 2008/01/01 and that the patient was also discharged from hospital. The final result produced a PADS2_PATIENTS_DIM table with 225,060 entries instead of the 1,598,380 entries of the full PADS2_PATIENTS_DIM table. The reason for this restriction was due to the limited processing quota allowed on the GRLS server for this experiment.

11.6.4.1. Deterministic Linkage

The PADS2_PATIENTS_DIM table was logically partitioned into five sets of groups to enable GRLS to perform the record linkage. The same five set groupings were used for both deterministic linkage and probabilistic linkage.

The deterministic linkage outcomes using Oracle SQL joins on different column combinations are summarized in table 11-14.

Table 11-14: Deterministic Linkage Outcomes for PADSDM

ORACLE SQL JOINS	Set 1	Set 2	Set 3	Set 4	Set 5	Total
	50,000	50,000	50,000	50,000	25,060	**225,060**
• Records with ID Number and length=13	16,761	15,012	14,642	15,757	8,243	**70,415**
• Duplicate records : Using Birth Date, Gender, Surname and First name	2,841	2,818	2,951	2,669	1,547	**12,826**
• Duplicate records : Using Birth Date, Gender, Surname and Initials	4,228	4,013	4,170	3,866	2,280	**18,557**
• Duplicate records: Using Birth Date, Gender, Surname, First name and Initials	2,661	2,661	2,779	2,519	1,453	**12,073**
• Duplicate records: Using Birth Date, Gender, Surname, First name, Initials and ID Number	650	609	566	610	336	**2,771**
• Duplicate records: Using ID number	1,692	1,535	1,463	1,654	853	**7,197**

11.6.4.2. Probabilistic Linkage

Rules and Weights

Table 11-15 outlines the rules and a short description of each rule that was created in GRLS for the probabilistic record matching.

Table 11-15: List of rules used in GRLS for PADSDM

Rule	Rule #	Rule Description
P_TOWN	50	Comparison on the town name of the patient
P_S_DIST	60	Comparison on the sub district name of the patient
P_DIST	70	Comparison on the district name of the patient
GENDER	80	Comparison on the gender
BIRTHDATE	100	Comparison on the date of birth
FSTNAME	110	Comparison on first name

Rule	Rule #	Rule Description
FS_NAME	130	Comparison on surname = first name and first name = surname
SURNAME	120	Comparison on surname
FS_NAME	130	Comparison on surname = first name and first name = surname
X_NAME	140	Comparison on first name and surname in case they are switched. Only execute this rule if SURNAME != A, and FSTNAME != A and FS_NAME != A
ADDRESS	150	Comparison on the street name
ADDRESS_NR	160	Comparison on the street number
MARRIED	170	Comparison on marital status
ID_NUMBER	180	Comparison on the ID Number (including allowing a disagree on 1 numeric value)

Decide Stage and Group Stage Actions

With the initial pairs of each data set available from the Search Stage, adjustments to the odds ratios and thresholds had to be made on an iterative manner. The original threshold of (-75, 25, 100) was adjusted until the number of possible and definitive links provided acceptable results. Tabulated below (see table 11-16) are the final outcomes using thresholds that was providing the maximum number of true DEFINITES.

Table 11-16: Outcomes of Decide and Group Stage for PADSDM

GRLS Outcomes	Set 1	Set 2	Set 3	Set 4	Set 5	Total
	50,000	50,000	50,000	50,000	25,060	**225,060**
Final Threshold	(50,105)	(50,105)	(50,103)	(50,103)	(50,103)	**-**
DEFINITE	4,100	3,804	3,960	4,010	2,109	**17,983**
POSSIBLE	1,419	1,716	3,112	1,451	911	**8,609**
REJECTED	743	660	4,471	955	717	**7,546**
Total Pairs	**6,262**	**6,180**	**11,543**	**6,416**	**3,737**	**34,138**

11.6.4.3. Linkage Findings Discussion

The most accurate multicolumn SQL join produced 12,073 duplicates entries. This represents a deterministic matching result on the following columns: Birth Date, Gender, Surname, First name and Initials. GRLS produced 17,983 DEFINITES pairs using probabilistic matching. When converting the DEFINITES pairs into individual row entries, GRLS identified 28,622 duplicate entries.

Using the newly identified duplicate entries (28,622) the dimension table had to be de-duplicated. The GRLS_PADS_PATIENTS_LINKED table containing 225,060 table rows was reduced to 209,560 table rows. This is a direct reduction of **15,500** table rows (**6.89%**). This de-duplicated table will be used for the two-file linkage which will be discussed later in the chapter.

11.6.5. HOSPDM

Both deterministic linkage and probabilistic linkage were performed on the patient dimension table (HOSP_PATIENTS_DIM) in the HOSP data mart. At the time of conducting this linkage experiment only data of 2008 was used. The selection criteria used were only admissions after 2008/01/01 and that the patient was also discharged from hospital. The final result produced a HOSP_PATIENTS_DIM table with 71,188 entries instead of the 654,446 entries of the full HOSP_PATIENTS_DIM table. The reason for this restriction was due to the limited processing quota allowed on the GRLS server for this experiment.

11.6.5.1. Deterministic Linkage

The HOSP_PATIENTS_DIM table was logically partitioned into two sets of groups to enable GRLS to perform the record linkage. The same two set groupings where used for both deterministic linkage and probabilistic linkage.

An important discovery was made when this data was closely scrutinized and analyzed. Validated ID numbers were only 12 digits long instead of the prescribed and standard 13 digits. A close investigation revealed that the Meditech Admission module was setup to only use 12 digits ID Numbers. SQL code was developed to generate the 13th digit, which is a control digit and can be calculated using an algorithm.

The deterministic linkage outcomes using Oracle SQL joins on different column combinations are summarized in table 11-17.

Table 11-17: Deterministic Linkage Outcomes for HOSPDM

ORACLE SQL JOINS	Set 1	Set 2	Total
	40,000	31,188	**71,188**
• Records with ID Number and length=13	16,552	13,137	**29,689**
• Duplicate records : Using Birth Date, Gender, Surname and First name	2,122	1,743	**3,865**
• Duplicate records : Using Birth Date, Gender, Surname and Initials	2,615	2,152	**4,767**
• Duplicate records: Using Birth Date, Gender, Surname, First name and Initials	2,122	1,743	**3,865**
• Duplicate records: Using Birth Date, Gender, Surname, First name, Initials and ID Number	964	775	**1,739**
• Duplicate records: Using ID number	1,379	1,109	**2,488**

11.6.5.2. Probabilistic Linkage

Rules and Weights

Table 11-18 outlines the rules and a short description of each rule that was created in GRLS for the probabilistic record matching. Because both the ARVDM and the HOSPDM are obtained from the Meditech system, the rules for the HOSPDM were almost identical to the rules used for the ARVDM.

Table 11-18: List of rules used in GRLS for HOSPDM

Rule	Rule #	Rule Description
P_TOWN	50	Comparison on the town name of the patient
P_S_DIST	60	Comparison on the sub district name of the patient
P_DIST	70	Comparison on the district name of the patient
GENDER	80	Comparison on the gender
BIRTHDATE	100	Comparison on the date of birth
FSTNAME	110	Comparison on first name
SURNAME	120	Comparison on surname
FS_NAME	130	Comparison on surname = first name and first name = surname
X_NAME	140	Comparison on first name and surname in case they are switched. Only execute this rule if SURNAME != A, and FSTNAME != A and FS_NAME != A
ADDRESS	150	Comparison on the street name
ADDRESS_NR	160	Comparison on the street number
MARRIED	170	Comparison on marital status
ID_NUMBER	180	Comparison on the ID Number (including allowing a disagree on 1 numeric value)

Decide Stage and Group Stage Actions

With the initial pairs of each data set available from the Search Stage, adjustments to the odds ratios and thresholds had to be made on an iterative manner. The original threshold of (-75, 25, 100) was adjusted until the number of possible and definitive links provided acceptable results. Tabulated below (see table 11-19) are the final outcomes using thresholds that was providing the maximum number of true DEFINITES.

Table 11-19: Outcomes of Decide and Group Stage for HOSPDM

GRLS Outcomes	Set 1	Set 2	**Total**
	40,000	31,188	**71,188**
Final Threshold	(50,95)	(50,90)	**-**
DEFINITE	1,543	1,326	**2,869**
POSSIBLE	897	697	**1,594**
REJECTED	3,875	2,139	**6,014**
Total Pairs	**6,315**	**4,162**	**10,477**

11.6.5.3. Linkage Findings Discussion

The most accurate multicolumn SQL join produced 3,865 duplicates entries. This represents a deterministic matching result on the following columns: Birth Date, Gender, Surname, First name and Initials. GRLS produced 2,869 DEFINITES pairs using probabilistic matching. When converting the DEFINITES pairs into individual row entries, GRLS identified 5,408 duplicate entries.

Using the newly identified duplicate entries (5,408) the dimension table had to be de-duplicated. The GRLS_HOSP_PATIENTS_LINKED table containing 71,188 table rows was reduced to 68,430 table rows. This is a direct reduction of **2,758** table rows (**3.87%**). This de-duplicated table will be used for the two-file linkage which will be discussed later in the chapter.

11.6.6. TBDM

Both deterministic linkage and probabilistic linkage was performed on the patient dimension table (TB_PATIENTS_DIM) in the TB data mart. At the time of conducting this linkage experiment only data of 2007 and 2008 were used. The selection criteria used was the treatment date after 2007/01/01. The final result produced a TB_PATIENTS_DIM table with 39,731 entries instead of the 128,086 entries of the full TB_PATIENTS_DIM table. The reason for this restriction was due to the limited processing quota allowed on the GRLS server for this experiment.

11.6.6.1. Deterministic Linkage

An important discovery was made when this data was closely scrutinized and analyzed. A large number of duplicate records had alternating genders, which was more than one would anticipate originating from capturing errors. In theory this meant that this data was actually in a poor state. This meant that the decision to only use 2007 and 2008 data was confirmed to be a wise one.

To improve the linkage exercise a calculated field called BIRTHYEAR was added to the dimension table. This field was calculated using the difference between age and registration date. All BIRTHYEAR values were set to the Year + ‘01/01’ as no other information on month or day was available.

The deterministic linkage outcomes using Oracle SQL joins on different column combinations are summarized in table 11-20.

Table 11-20: Deterministic Linkage Outcomes for TBDM

ORACLE SQL JOINS	Set 1
• Duplicate records : Using Birth Year, Gender, Surname and First name	3,897
• Duplicate records : Using Birth Year, Gender, Surname and Initials	5,913
• Duplicate records: Using Birth Year, Gender, Surname, First name and Initials	3,897

11.6.6.2. Probabilistic Linkage

Rules and Weights

Table 11-21 outlines the rules and a short description of each rule that was created in GRLS for the probabilistic record matching.

Table 11-21: List of rules used in GRLS for TBDM

Rule	Rule #	Rule Description (to review pairs only)
F_FACILITY	40	Comparison on the name of the facility the patient visited
F_TOWN	50	Comparison on the town name of the facility the patient visited
F_S_DIST	60	Comparison on the sub district name of the facility the patient visited
F_DIST	70	Comparison on the district name of the facility the patient visited
Rule	Rule #	Rule Description (to match pairs only)
GENDER	80	Comparison on the gender
BIRTHYEAR	100	Comparison on the year of birth
FSTNAME	110	Comparison on first name
SURNAME	120	Comparison on surname
FS_NAME	130	Comparison on surname = first name and first name = surname
X_NAME	140	Comparison on first name and surname in case they are switched. Only execute this rule if SURNAME != A, and FSTNAME != A and FS_NAME != A

Decide Stage and Group Stage Actions

With the initial pairs of each data set available from the Search Stage, adjustments to the odds ratios and thresholds had to be made on an iterative manner. The original threshold of (-75, 25, 100) was adjusted until the number of possible and definitive links provided acceptable results. Tabulated below (see table 11-22) are the final outcomes using thresholds that was providing the maximum number of true DEFINITES.

Table 11-22: Outcomes of Decide and Group Stage for TBDM

GRLS Outcomes	Set 1
	39,731
Final Threshold	(50,75)
DEFINITE	3,352
POSSIBLE	4,140
REJECTED	4,576
Total Pairs	**12,068**

11.6.6.3. Linkage Findings Discussion

The most accurate multicolumn SQL join produced 3,897 duplicates entries. This represents a deterministic matching result on the following columns: Birth Year, Gender, Surname, First name and Initials. GRLS produced 3,352 DEFINITES pairs using probabilistic matching. When converting the DEFINITES pairs into individual row entries GRLS identified 5,716 duplicate entries. A low number of results were observed, due to poor data quality. The majority of data problems reside within the GENDER field where many records of the same patient had alternating gender values.

Using the newly identified duplicate entries (5,716) the dimension table had to be de-duplicated. The GRLS_TB_PATIENTS_LINKED table containing 39,731 table rows was reduced to 36,711 table rows. This is a direct reduction of **3,020** table rows (**7.60%**). This de-duplicated table will be used for the two-file linkage which will be discussed later in the chapter.

11.6.7. Internal Record Linkage Summary

The GRLS internal linkage outcomes are summarized in table 11-23. These newly created de-duplicated dimension tables will be used for the two-file record linkage, which will be discussed in the next section.

Table 11-23: Outcomes of Internal Linkage using Probabilistic Record Matching

	NOTIF DM	NHLS DM	ARV DM	PADS DM	HOSP DM	TB DM
Number of rows (before using GRLS to de-duplicate)	11,445	94,810	65,300	225,060	71,188	39,731
Number of rows (after using GRLS to de-duplicate)	10,164	58,378	63,944	209,560	68,430	36,711
Duplicate rows removed	**1,281**	**36,432**	**1,356**	**15,500**	**2,758**	**3,020**
Duplicate rows removed (%)	**▼ 11.19%**	**▼ 38.42%**	**▼ 2.07%**	**▼ 6.89%**	**▼ 3.87%**	**▼ 7.60%**

11.7. Two-file Record Linkage

The following section will describe the two-file linkage that was performed between the different data marts and the ARVDM and the lessons learned.

11.7.1. Mapped Patient Table

Earlier in the chapter the usage of a proposed mapping table was discussed to link all the patient dimension tables together. Depicted in figure 11-12 is the layout of the proposed mapping table. As the two-file linkage experiments between the data marts concluded, the relevant columns were updated to provide the linkage.

COLUMN ID	Column Name	Data Type
1	ARV_PATIENT_KEY	VARCHAR2(20 BYTE)
2	ARV_GRLS_NEWID	NUMBER
3	IDNUMBER	VARCHAR2(13 BYTE)
4	SURNAME	VARCHAR2(50 BYTE)
5	FIRSTNAME	VARCHAR2(50 BYTE)
6	INITIALS	VARCHAR2(5 BYTE)
7	ARV_ENABLED	CHAR(1 BYTE)
8	NOTIF_GRLS_NEWID	NUMBER
9	NHLS_GRLS_NEWID	NUMBER
10	HOSP_GRLS_NEWID	NUMBER
11	PADS_GRLS_NEWID	NUMBER
12	TB_GRLS_NEWID	NUMBER

Figure 11-12: PATIENTS_MAPPED_DIM mapping table

11.7.2. Data Quality

An additional field rule was added to match on INITIALS between the ARVDM and the different data marts. The reason for adding an additional rule was because **17,712** out of the **63,944** records in the GRLS_ARV_PATIENTS_LINKED table had no first name. Only **94** out of the **63,944** rows had no initials. Adding initials would subsequently increase the probabilistic matching accuracy.

11.7.3. NDDM with ARVDM

Both deterministic linkage and probabilistic linkage were performed between the GRLS_ARV_PATIENTS_LINKED and GRLS_NOTIF_PATIENTS_LINKED. Both tables contained ***de-duplicated*** entries and had 63,944 and 10,164 rows respectively at the time of this experiment.

11.7.3.1. Deterministic Linkage

An additional calculated field was added to GRLS_ARV_PATIENTS_LINKED that contained the birth year and '0101'. This field was added to match up with the calculated BIRTHYEAR field that was added to GRLS_NOTIF_PATIENTS_LINKED during the internal linkage experiments.
The deterministic linkage outcomes using Oracle SQL joins on different column combinations are summarized in table 11-24.

Table 11-24: Deterministic Linkage Outcomes for ARVDM and NOTIFDM

ORACLE SQL JOINS	Set 1 (ARVDM)	Set 2 (NOTIFDM)
	63,944	10,164
• Matched records : Using Birth Year, Gender, Surname and First name	227	
• Matched records : Using Birth Year, Gender, Surname and Initials	661	
• Matched records: Using Birth Year, Gender, Surname, First name and Initials	215	

11.7.3.2. Probabilistic Linkage

Rules and Weights

Table 11-25 outlines the rules and a short description of each rule that was created in GRLS for the probabilistic record matching. The weights of the rules are outlined in table 11-26.

Table 11-25: List of rules used in GRLS for ARVDM and NOTIFDM

Rule	Rule #	Rule Description (to review pairs only)
P_TOWN	50	Comparison on the town name of the patient
F_S_DIST	60	Comparison on the sub district name of the town of the patient
F_DIST	70	Comparison on the district name of the town of the patient
GENDER	80	Comparison on the gender
BIRTHYEAR	90	Comparison on the year of birth
INITIALS	100	Comparison on initials
FSTNAME	110	Comparison on first name
SURNAME	120	Comparison on surname
FS_NAME	130	Comparison on surname = first name and first name = surname
X_NAME	140	Comparison on first name and surname in case they are switched. Only execute this rule if SURNAME != A, and FSTNAME != A and FS_NAME != A
ADDRESS	150	Comparison on the street name
ADDRESS_NR	160	Comparison on the street number

Table 11-26: Rules with their respective weights in GRLS for ARVDM and NOTIFDM

rule	rule#	outcome	weight linked	weight non-linked	ORw
P_TOWN	50	D	-20	2	-18
P_TOWN	50	A	-4	37	33
P_S_DIST	60	D	-25	3	-22
P_S_DIST	60	A	-3	33	30
P_DIST	70	D	-33	4	-29
P_DIST	70	A	-2	23	21
GENDER	80	D	-66	10	-56
GENDER	80	A	0	10	10
BIRTHYEAR	90	D	-100	5	-95
BIRTHYEAR	90	A (Y1)	-5	20	15
BIRTHYEAR	90	A (Y2)	-15	0	-15
BIRTHYEAR	90	A (Y3)	-30	0	-30
INITIALS	100	D	-20	5	-15
INITIALS	100	A (C1)	-5	20	15
INITIALS	100	A (T1)	-8	18	10
FSTNAME	110	D	-35	5	-30
FSTNAME	110	A (C1)	-10	40	30
FSTNAME	110	A (S1)	-8	25	17
FSTNAME	110	A (T1)	0	10	10
FSTNAME	110	A (X1)	0	15	15
FSTNAME	110	A (P1)	0	5	5
SURNAME	120	D	-55	5	-50
SURNAME	120	A (C1)	-20	50	30
SURNAME	120	A (S1)	-8	30	22
SURNAME	120	A (T1)	0	10	10
SURNAME	120	A (X1)	0	15	15
SURNAME	120	A (P1)	0	5	5
FS_NAME	130	D	-18	15	-3
FS_NAME	130	A	-10	45	35
X_NAME	140	D	0	0	0
X_NAME	140	A	-10	40	30
ADRESS	150	D	-5	4	-1
ADRESS	150	A (C1)	-2	27	25
ADRESS	150	A (S1)	-5	13	8
ADRESS	150	A (X1)	-3	16	13
ADRESS	150	A (P1)	-7	11	4
ARDESSNR	160	D	-5	4	-1
ARDESSNR	160	A	-2	12	10

Decide Stage and Group Stage Actions

With the initial pairs of each data set available from the Search Stage, adjustments to the odds ratios and thresholds had to be made on an iterative manner. The original threshold of (-75, 25, 100) was adjusted until the number of possible and definitive links provided acceptable results. Tabulated below (see Table 11-27) are the final outcomes using thresholds that was providing the maximum number of true DEFINITES (after grouping and mapping of one-to-one was done)

Table 11-27: Outcomes of Decide and Group Stage for ARVDM and NOTIFDM

GRLS Outcomes	Set 1 (ARVDM)	Set 2 (NOTIFDM)
	63,944	10,164
Final Threshold	(50,115)*	
DEFINITE	894	
POSSIBLE	8,571	
REJECTED	18,887	
Total Pairs	**28,352**	

* The definite threshold of 115 is calculated on the basis that the initials (+15), gender (+10), surname (+30), address (+25), district (+20) and birth year (+15) columns have agreements.

11.7.3.3. Linkage Findings Discussion

The most accurate multicolumn SQL joins produced respectively 661 (without first name) and 227 (with first name) matches. GRLS produced 894 DEFINITE pairs using all the rules and weights as defined in tables 11.25 and 11.26. Using the DEFINITE pairs, the mapping table was updated to reflect the linkage between patients in the ARVDM and NOTIFDM.

11.7.4. NHLSDM with ARVDM

Both deterministic linkage and probabilistic linkage were performed between the GRLS_ARV_PATIENTS_LINKED and GRLS_NHLS_PATIENTS_LINKED. Both tables contained ***de-duplicated*** entries and had 63,944 and 58,378 rows respectively at the time of this experiment.

11.7.4.1. Deterministic Linkage

The deterministic linkage outcomes using Oracle SQL joins on different column combinations are summarized in table 11-28.

Table 11-28: Deterministic Linkage Outcomes for ARVDM and NHLSDM

ORACLE SQL JOINS	Set 1 (ARVDM)	Set 2 (NHLSDM)
	63,944	58,378
• Matched records : Using ID Number	2,905	
• Matched records : Using Birth Date, Gender, Surname and First name	3,858	
• Matched records : Using Birth Date, Gender, Surname and Initials	7,460	
• Matched records: Using Birth Date, Gender, Surname, First name and Initials	3,551	
• Matched records: Using Birth Date, Gender, Surname, First name, Initials and ID Number	400	

11.7.4.2. Probabilistic Linkage

Rules and Weights

Table 11-29 outlines the rules and a short description of each rule that was created in GRLS for the probabilistic record matching. The weights of the rules are outlined in table 11-30.

Table 11-29: List of rules used in GRLS for ARVDM and NHLSDM

Rule	Rule #	Rule Description (to match pairs only)
ID_NUMBER	70	Comparison on the ID Number (including allowing a disagreement on 1 numeric value)
GENDER	80	Comparison on the gender
BIRTHDATE	90	Comparison on the date of birth
INITIALS	100	Comparison on initials
FSTNAME	110	Comparison on first name
SURNAME	120	Comparison on surname
FS_NAME	130	Comparison on surname = first name and first name = surname
X_NAME	140	Comparison on first name and surname in case they are switched. Only execute this rule if SURNAME != A, and FSTNAME != A and FS_NAME != A

Table 11-30: Rules with their respective weights in GRLS for ARVDM and NHLSDM

rule	rule#	outcome	weight linked	weight non-linked	ORw
ID_NUMBER	70	D	-100	5	-95
ID_NUMBER	70	A (C1)	-5	100	95
		A (T1)	0	30	30
GENDER	80	D	-66	100	-56
GENDER	80	A	0	10	10

rule	rule#	outcome	weight linked	weight non-linked	ORw
BIRTHDATE	90	D	-100	5	-95
BIRTHDATE	90	A (Y1)	-5	20	15
BIRTHDATE	90	A (Y2)	-15	0	-15
BIRTHDATE	90	A (Y3)	-30	0	-30
INITIALS	100	D	-20	5	-15
INITIALS	100	A (C1)	-5	20	15
INITIALS	100	A (T1)	-8	18	10
FSTNAME	110	D	-35	5	-30
FSTNAME	110	A (C1)	-10	40	30
FSTNAME	110	A (S1)	-8	25	17
FSTNAME	110	A (T1)	0	10	10
FSTNAME	110	A (X1)	0	15	15
FSTNAME	110	A (P1)	0	5	5
SURNAME	120	D	-55	5	-50
SURNAME	120	A (C1)	-20	50	30
SURNAME	120	A (S1)	-8	30	22
SURNAME	120	A (T1)	0	10	10
SURNAME	120	A (X1)	0	15	15
SURNAME	120	A (P1)	0	5	5
FS_NAME	130	D	-18	15	-3
FS_NAME	130	A	-10	45	35
X_NAME	140	D	0	0	0
X_NAME	140	A	-10	40	30

Decide Stage and Group Stage Actions

With the initial pairs of each data set available from the Search Stage, adjustments to the odds ratios and thresholds had to be made on an iterative manner. The original threshold of (-75, 25, 100) was adjusted until the number of possible and definitive links provided acceptable results. Tabulated below (see Table 11-31) are the final outcomes using thresholds that was providing the maximum number of true DEFINITES (after grouping and mapping of one-to-one was done)

Table 11-31: Outcomes of Decide and Group Stage for ARVDM and NHLSDM

GRLS Outcomes	Set 1 (ARVDM)	Set 2 (NHLSDM)
	63,944	58,378
Final Threshold	(50,70)*	
DEFINITE	14,611	
POSSIBLE	8,052	
REJECTED	5,309	
Total Pairs	**27,972**	

* The definite threshold of 70 is calculated on the basis that the initials (+15), gender (+10), surname (+30) and birth date (+15) columns have agreements.

11.7.4.3. Linkage Findings Discussion

The most accurate multicolumn SQL joins produced respectively 7,460 (without first name) and 3,858 (with first name) matches. GRLS produced 14,611 DEFINITE pairs using all the rules and weight as defined in tables 11-29 and 11-30. Using the DEFINITE pairs, the mapping table was updated to reflect the linkage between patients in the ARVDM and NHLSDM.

11.7.5. HOSPDM with ARVDM

Both deterministic linkage and probabilistic linkage were performed between the GRLS_ARV_PATIENTS_LINKED and GRLS_HOSP_PATIENTS_LINKED. Both tables contained ***de-duplicated*** entries and had 63,944 and 68,430 rows respectively at the time of this experiment.

11.7.5.1. Deterministic Linkage

The deterministic linkage outcomes using Oracle SQL joins on different column combinations are summarized in table 11-32.

Table 11-32: Deterministic Linkage Outcomes for ARVDM and HOSPDM

ORACLE SQL JOINS	Set 1 (ARVDM)	Set 2 (HOSPDM)
	63,944	68,430
• Matched records: Using ID Number	1,703	
• Matched records : Using Birth Date, Gender, Surname and First name	1,379	
• Matched records : Using Birth Date, Gender, Surname and Initials	1,971	

• Matched records: Using Birth Date, Gender, Surname, First name and Initials	1,312
• Matched records: Using Birth Date, Gender, Surname, First name, Initials and ID Number	939

11.7.5.2. Probabilistic Linkage

Rules and Weights

Table 11-33 outlines the rules and a short description of each rule that was created in GRLS for the probabilistic record matching. The weights of the rules are outlined in table 11-34.

Table 11-33: List of rules used in GRLS for ARVDM and HOSPDM

Rule	Rule #	Rule Description (to match pairs only)
P_TOWN	50	Comparison on the town name of the patient
P_S_DIST	60	Comparison on the sub district name of the patient
P_DIST	70	Comparison on the district name of the patient
GENDER	80	Comparison on the gender
BIRTHDATE	90	Comparison on the date of birth
INITIALS	100	Comparison on initials
FSTNAME	110	Comparison on first name
SURNAME	120	Comparison on surname
FS_NAME	130	Comparison on surname = first name and first name = surname
X_NAME	140	Comparison on first name and surname in case they are switched. Only execute this rule if SURNAME != A, and FSTNAME != A and FS_NAME != A
ADDRESS	150	Comparison on the street name
ADDRESS_NR	160	Comparison on the street number
MARRIED	170	Comparison on marital status
ID_NUMBER	180	Comparison on the ID Number (including allowing a disagree on 1 numeric value)

Table 11-34: Rules with their respective weights in GRLS for ARVDM and HOSPDM

rule	rule#	outcome	weight linked	weight non-linked	Orw
P_TOWN	50	D	-20	2	-18
P_TOWN	50	A	-4	37	33
P_S_DIST	60	D	-25	3	-22
P_S_DIST	60	A	-3	33	30
P_DIST	70	D	-33	4	-29
P_DIST	70	A	-2	23	21
GENDER	80	D	-66	10	-56
GENDER	80	A	0	10	10

rule	rule#	outcome	weight linked	weight non-linked	ORw
BIRTHDATE	90	D	-100	5	-95
BIRTHDATE	90	A (Y1)	-5	20	15
BIRTHDATE	90	A (Y2)	-15	0	-15
BIRTHDATE	90	A (Y3)	-30	0	-30
INITIALS	100	D	-20	5	-15
INITIALS	100	A (C1)	-5	20	15
INITIALS	100	A (T1)	-8	18	10
FSTNAME	110	D	-35	5	-30
FSTNAME	110	A (C1)	-10	40	30
FSTNAME	110	A (S1)	-8	25	17
FSTNAME	110	A (T1)	0	10	10
FSTNAME	110	A (X1)	0	15	15
FSTNAME	110	A (P1)	0	5	5
SURNAME	120	D	-55	5	-50
SURNAME	120	A (C1)	-20	50	30
SURNAME	120	A (S1)	-8	30	22
SURNAME	120	A (T1)	0	10	10
SURNAME	120	A (X1)	0	15	15
SURNAME	120	A (P1)	0	5	5
FS_NAME	130	D	-18	15	-3
FS_NAME	130	A	-10	45	35
X_NAME	140	D	0	0	0
X_NAME	140	A	-10	40	30
ADRESS	150	D	-5	4	-1
ADRESS	150	A (C1)	-2	27	25
ADRESS	150	A (S1)	-5	13	8
ADRESS	150	A (X1)	-3	16	13
ADRESS	150	A (P1)	-7	11	4
ARDESSNR	160	D	-5	4	-1
ARDESSNR	160	A	-2	12	10
MARRIED	170	D	-35	5	-30
MARRIED	170	A	-5	20	15
ID_NUMBER	180	D	-100	5	-95
ID_NUMBER	180	A (C1)	-5	100	95
		A (T1)	0	30	30

Decide Stage and Group Stage Actions

With the initial pairs of each data set available from the Search Stage, adjustments to the odds ratios and thresholds had to be made on an iterative manner. The original threshold of (-75, 25, 100) was adjusted until the number of possible and definitive links provided acceptable results. Tabulated below (see Table 11-35) are the final outcomes using thresholds that was providing the maximum number of true DEFINITES (after grouping and mapping of one-to-one was done)

Table 11-35: Outcomes of Decide and Group Stage for ARVDM and HOSPDM

GRLS Outcomes	Set 1 (ARVDM)	Set 2 (HOSPDM)
	63,944	68,430
Final Threshold	(60,100)*	
DEFINITE	2,405	
POSSIBLE	467	
REJECTED	889	
Total Pairs	**3,758**	

* The definite threshold of 100 is calculated on the basis that the initials (+15), gender (+10), surname (+30), birth date (+15) and other (+30) columns have agreements.

11.7.5.3. Linkage Findings Discussion

The most accurate multicolumn SQL joins produced respectively 1,971 (without first name) and 1,379 (with first name) matches. GRLS produced 2,405 DEFINITE pairs using all the rules and weights as defined in tables 11-33 and 11-34. Using the DEFINITE pairs, the mapping table was updated to reflect the linkage between patients in the ARVDM and HOSPDM.

11.7.6. PADSDM with ARVDM

Both deterministic linkage and probabilistic linkage were performed between the GRLS_ARV_PATIENTS_LINKED and GRLS_PADS_PATIENTS_LINKED. Both tables contained ***de-duplicated*** entries and had 63,944 and 209,560 rows respectively at the time of this experiment.

11.7.6.1. Deterministic Linkage

The deterministic linkage outcomes using Oracle SQL joins on different column combinations are summarized in table 11-36.

Table 11-36: Deterministic Linkage Outcomes for ARVMD and PADSDM

ORACLE SQL JOINS	Set 1 (ARVDM)	Set 2 (PADSDM)
	63,944	209,560
• Matched records: Using ID Number	3,786	
• Matched records : Using Birth Date, Gender, Surname and First name	2,242	
• Matched records : Using Birth Date, Gender, Surname and Initials	4,092	
• Matched records: Using Birth Date, Gender, Surname, First name and Initials	2,093	
• Matched records: Using Birth Date, Gender, Surname, First name, Initials and ID Number	1,231	

11.7.6.2. Probabilistic Linkage

Rules and Weights

Table 11-37 outlines the rules and a short description of each rule that was created in GRLS for the probabilistic record matching. The weights of the rules are outlined in table 11-38.

Table 11-37: List of rules used in GRLS for ARVMD and PADSDM

Rule	Rule #	Rule Description (to match pairs only)
P_TOWN	50	Comparison on the town name of the patient
P_S_DIST	60	Comparison on the sub district name of the patient
P_DIST	70	Comparison on the district name of the patient
GENDER	80	Comparison on the gender
BIRTHDATE	90	Comparison on the date of birth
INITIALS	100	Comparison on the initials
FSTNAME	110	Comparison on first name
SURNAME	120	Comparison on surname
FS_NAME	130	Comparison on surname = first name and first name = surname
X_NAME	140	Comparison on first name and surname in case they are switched. Only execute this rule if SURNAME != A, and FSTNAME != A and FS_NAME != A
ADDRESS	150	Comparison on the street name
ADDRESS_NR	160	Comparison on the street number
ID_NUMBER	170	Comparison on the ID Number (including allowing a disagree on 1 numeric value)

Table 11-38: Rules with their respective weights in GRLS for ARVMD and PADSDM

rule	rule#	outcome	weight linked	weight non-linked	ORw
P_TOWN	50	D	-20	2	-18
P_TOWN	50	A	-4	37	33
P_S_DIST	60	D	-25	3	-22
P_S_DIST	60	A	-3	33	30
P_DIST	70	D	-33	4	-29
P_DIST	70	A	-2	23	21
GENDER	80	D	-66	10	-56
GENDER	80	A	0	10	10
BIRTHDATE	90	D	-100	5	-95
BIRTHDATE	90	A (Y1)	-5	20	15
BIRTHDATE	90	A (Y2)	-15	0	-15
BIRTHDATE	90	A (Y3)	-30	0	-30
INITIALS	100	D	-20	5	-15
INITIALS	100	A (C1)	-5	20	15
INITIALS	100	A (T1)	-8	18	10
FSTNAME	110	D	-35	5	-30
FSTNAME	110	A (C1)	-10	40	30
FSTNAME	110	A (S1)	-8	25	17
FSTNAME	110	A (T1)	0	10	10
FSTNAME	110	A (X1)	0	15	15
FSTNAME	110	A (P1)	0	5	5
SURNAME	120	D	-55	5	-50
SURNAME	120	A (C1)	-20	50	30
SURNAME	120	A (S1)	-8	30	22
SURNAME	120	A (T1)	0	10	10
SURNAME	120	A (X1)	0	15	15
SURNAME	120	A (P1)	0	5	5
FS_NAME	130	D	-18	15	-3
FS_NAME	130	A	-10	45	35
X_NAME	140	D	0	0	0
X_NAME	140	A	-10	40	30
ADRESS	150	D	-5	4	-1
ADRESS	150	A (C1)	-2	27	25
ADRESS	150	A (S1)	-5	13	8
ADRESS	150	A (X1)	-3	16	13
ADRESS	150	A (P1)	-7	11	4

rule	rule#	outcome	weight linked	weight non-linked	ORw
ARDESSNR	160	D	-5	4	-1
ARDESSNR	160	A	-2	12	10
ID_NUMBER	170	D	-100	5	-95
ID_NUMBER	170	A (C1)	-5	100	95
		A (T1)	0	30	30

Decide Stage and Group Stage Actions

With the initial pairs of each data set available from the Search Stage, adjustments to the odds ratios and thresholds had to be made on an iterative manner. The original threshold of (-75, 25, 100) was adjusted until the number of possible and definitive links provided acceptable results. Tabulated below (see Table 11-39) are the final outcomes using thresholds that was providing the maximum number of true DEFINITES (after grouping and mapping of one-to-one was done)

Table 11-39: Outcomes of Decide and Group Stage for ARVDM and PADSDM

GRLS Outcomes	Set 1 (ARVDM)	Set 2 (PADSDM)
	63,944	209,560
Final Threshold	(60,95)*	
DEFINITE	6,686	
POSSIBLE	1,699	
REJECTED	3,800	
Total Pairs	**12,185**	

* The definite threshold of 95 is calculated on the basis that the initials (+15), gender (+10), surname (+30), birth date (+15) and other (+25) columns have agreements.

11.7.6.3. Linkage Findings Discussion

The most accurate multicolumn SQL joins produced respectively 4,092 (without first name) and 2,242 (with first name) matches. GRLS produced 6,686 DEFINITE pairs using all the rules and weights as defined in tables 11-37 and 11-38. Using the DEFINITE pairs, the mapping table was updated to reflect the linkage between patients in the ARVDM and PADSDM.

11.7.7. TBDM with ARVDM

Both deterministic linkage and probabilistic linkage were performed between the GRLS_ARV_PATIENTS_LINKED and GRLS_TB_PATIENTS_LINKED. Both tables contained ***de-duplicated*** entries and had 63,944 and 36,711 rows respectively at the time of this experiment. All the data was used for the matching.

11.7.7.1. Deterministic Linkage

An additional calculated field was added to GRLS_ARV_PATIENTS_LINKED that contained the birth year and '0101'. This field was added to match up with the calculated BIRTHYEAR field that was added to GRLS_TB_PATIENTS_LINKED during the internal linkage experiments.

The deterministic linkage outcomes using Oracle SQL joins on different column combinations are summarized in table 11-40.

Table 11-40: Deterministic Linkage Outcomes for ARVDM and TBDM

ORACLE SQL JOINS	Set 1 (ARVDM)	Set 2 (TBDM)
	63,944	36,711
• Matched records : Using Birth Year, Gender, Surname and First name	555	
• Matched records : Using Birth Year, Gender, Surname and Initials	2,525	
• Matched records: Using Birth Year, Gender, Surname, First name and Initials	510	

11.7.7.2. Probabilistic Linkage

Rules and Weights

Table 11-41 outlines the rules and a short description of each rule that was created in GRLS for the probabilistic record matching. The weights of the rules are outlined in table 11-42.

Table 11-41: List of rules used in GRLS for ARVDM and TBDM

Rule	Rule #	Rule Description (to match pairs only)
BIRTHYEAR	80	Comparison on the year of birth
GENDER	90	Comparison on the gender
INITIALS	100	Comparison on initials
FSTNAME	110	Comparison on first name
SURNAME	120	Comparison on surname
FS_NAME	130	Comparison on surname = first name and first name = surname
X_NAME	140	Comparison on first name and surname in case they are switched. Only execute this rule if SURNAME != A, and FSTNAME != A and FS_NAME != A

Table 11-42: Rules with their respective weights in GRLS for ARVDM and TBDM

rule	rule#	outcome	weight linked	weight non-linked	ORw
BIRTHYEAR	80	D	-100	5	-95
BIRTHYEAR	80	A (Y1)	-5	20	15
BIRTHYEAR	80	A (Y2)	-15	0	-15
BIRTHYEAR	80	A (Y3)	-30	0	-30
GENDER	90	D	-66	100	-56
GENDER	90	A	0	10	10
INITIALS	100	D	-20	5	-15
INITIALS	100	A (C1)	-5	20	15
INITIALS	100	A (T1)	-8	18	10
FSTNAME	110	D	-35	5	-30
FSTNAME	110	A (C1)	-10	40	30
FSTNAME	110	A (S1)	-8	25	17
FSTNAME	110	A (T1)	0	10	10
FSTNAME	110	A (X1)	0	15	15
FSTNAME	110	A (P1)	0	5	5
SURNAME	120	D	-55	5	-50
SURNAME	120	A (C1)	-20	50	30
SURNAME	120	A (S1)	-8	30	22
SURNAME	120	A (T1)	0	10	10
SURNAME	120	A (X1)	0	15	15
SURNAME	120	A (P1)	0	5	5
FS_NAME	130	D	-18	15	-3
FS_NAME	130	A	-10	45	35
X_NAME	140	D	0	0	0
X_NAME	140	A	-10	40	30

Decide Stage and Group Stage Actions

With the initial pairs of each data set available from the Search Stage, adjustments to the odds ratios and thresholds had to be made on an iterative manner. The original threshold of (-25, 25, 100) was adjusted until the number of possible and definitive links provided acceptable results. Tabulated below (see Table 11-43) are the final outcomes using thresholds that was providing the maximum number of true DEFINITES (after grouping and mapping of one-to-one was done)

Table 11-43: Outcomes of Decide and Group Stage for ARVDM and TBDM

GRLS Outcomes	Set 1 (ARVDM)	Set 2 (TBDM)
	63,944	36,711
Final Threshold	(70,85)*	
DEFINITE	1,557	
POSSIBLE	5,588	
REJECTED	88,703	
Total Pairs	**95,848**	

* The definite threshold of 85 is calculated on the basis that the initials with one typo (+10), gender (+10), surname with one typo (+30), first name with one typo (+10), birth year (+15) and other (+10) columns have agreements.

11.7.7.3. Linkage Findings Discussion

The most accurate multicolumn SQL joins produced respectively 2,525 (without first name) and 555 (with first name) matches. GRLS produced 1,557 DEFINITE pairs using all the rules and weights as defined in tables 11-41 and 11-42. A relatively low number of results were observed, due to poor data quality. The majority of data problems reside within the GENDER field where many records of the same patient had alternating gender values.

Although the multicolumn SQL join of 2,525 (without first name) produced the highest match results, it is not a reflection of the truth. The argument can be made that a substantial higher probability exists to match individuals using their initials, but it will produced a large number of *falsely matched* pairs. For example both first names MARIE and MARTA have the initial M, but clearly it indicates two different individuals. The first name is the important field and the key differentiator in this scenario.

Using the DEFINITE pairs, the mapping table was updated to reflect the linkage between patients in the ARVDM and TBDM.

11.7.8. Two-file Record Linkage Summary

By observing the summary of all the two-file record linkage experiments (see table 17.44) it is evident that probabilistic record matching outperformed deterministic record matching. All the probabilistic linkage results were merged into the mapping table, which will form the backbone of constructing the longitudinal record.

Table 11-44: Summary of the two-file linkage outcomes

	NOTIFDM	NHLSDM	PADSDM	HOSPDM	TBDM
De-duplicated entries	10,164	58,378	209,560	68,430	36,711
Deterministic matches to ARVDM (including first names)	227	7,460	2,242	1,379	555
Probabilistic matches to ARVDM (including first names)	894	14,611	6,686	2,405	1,557

11.8. Chapter Summary

This chapter examined and documented the use of probabilistic methods in linking up the relevant data fields from the individual data marts to provide a framework for a coherent longitudinal electronic patient record (LPR). Both deterministic and probabilistic linkage methods were employed on each data mart individually, and then between all the data marts and the ARV data mart. For all the experiments, probabilistic linkage performed better than deterministic linkage. By using this finding, only probabilistic linkage results were merged into the mapping table. The following chapter will outline the development and implementation of an algorithm to construct the LPR. The mapping table will play a turnkey role in this implementation.

CHAPTER 12 - SUPPLY STRATEGIC ARV INFORMATION FROM A LONGITUDINAL PATIENT RECORD

12.1. Introduction

This chapter will outline the implementation of an algorithm that will use the probabilistic linkage results that was obtained in the previous chapter. The end result will be a prototype, yet functional, LPR, that will be accessible using a web based application as user interface. This chapter fits into the *action taking* phase of the action research cycle.

12.2. Background

It is important to note that the FSDOH team not only needed an enterprise data warehouse to provide management with the necessary decision support, but also ***a longitudinal record of a patient***. This record will provide valuable input in changing policies, such as the implementation of antiretrovirals. In order to meet this goal, a uniformed (consolidated and conformed) patient dimension was necessary to link all the fact tables (in each data mart) and in theory produce a coherent and complete history of treatment or clinical encounters. This complete picture could then assist in the study of interdependencies and effects of other clinical encounters on the effectiveness of antiretrovirals.

In the OLTP world, this problem is also under the microscope and it is often referred to as the integrated Electronic Patient Record (EPR). An Electronic Patient Record is the ***longitudinal record of a patient*** that is captured in an online OLTP system. This OLTP system enables real-time online electronic order entry, documentation, results review and clinical decision support (Ebidia, Mulder, Tripp and Morgan, 1999). According to Ebidia et al. (1999) a patient's EPR should contain the following information:

- Demographic Information (incl. Personal Details)
- Past Medical History (incl. Allergies and Blood group)
- Medication Profile and Prescriptions
- Blood Test Results
- Imaging Results

Eichhorst (2002) argued that the the longitudinal record is more than a clinical repository and should also merge scheduling, billing, coverage and demographic data. It is, however, important to note that most of the work done on the LPR is within the OLTP environment. Eichhorst (2002) states that the patient-centric LPR must be the result of an integrated approach with registration, scheduling, clinical and billing systems working together in a single-application framework. Users should be able to logon

once to a computer workspace custom-tailored to their organizational role and access this wealth of discrete data. Individual organizations should be in a position to analyze their outcomes and begin to build their own best practice guidelines, contributing to the growing mass of medical knowledge in an actionable way (Eichhorst, 2002).

The FSDOH EPR (Electronic Patient Record) system is embedded within Meditech's Medical Record Interface (MRI) module. This module provides the patient demographic details and is referenced by all the Meditech modules, including MPM. The MPM module is used for capturing all the information on the patients taking part in the ART programme. This EPR is, however, OLTP based and is implemented on MUMPS technology. According to Ebidia et al. (1999) MUMPS is very efficient as a transaction processor. The problem with this OLTP based EPR is that it is not designed with decision-making in mind and according to Ebidia et al. (1999) lacks the necessary OLAP functionality. In addition to these shortcomings the OLTP based EPR also contains no information on TB encounters, NHLS blood results, hospital visits from both hospital information systems and linkage with the Department of Home Affairs.

According to (DeGruy, 2000) many health providers are migrating toward the use of computer-based patient records (CPRs). Carter (2001:9) defines the CPR as an electronic patient record that resides in a system designed to support users through availability of complete and accurate data, practitioner reminders and alerts, clinical decision support systems, links to bodies of medical knowledge and other aids. Carter (2001:8) refers to the work done by Dick et al. (1991) who stated that the CPR is a representation of all of a patient's data that one would find in the paper-based record, but in a coded and structured, machine-readable form. Carter (2001:8) quoted Dick et al. (1991) by mentioning that the Electronic Medical Records (EMR) and Electronic Patient Records (EPR) are in fact reasonable synonymous since both are electronic, machine-readable versions of data found in paper-based records, comprising both structured and unstructured patient data from disparate, computerized ancillary systems and document imaging systems. Carter (2001:9) concludes with the consideration that the EMR or EPR is transitional between the paper-based record and the CPR. A true CPR requires that every data item be uniquely coded and individually searchable; an EMR or EPR does not (Carter, 2001:9).

Rishel, Thomas, Handler and Edwards (2005) introduced the term Electronic Health Record (EHR) and defined it as an aggregation of patient-centric health data that originates in the patient record systems of multiple independent healthcare organizations for the purpose of facilitating care across multiple organizations. In other words the EHR is a long-term record for a patient, transcending his or her involvement with individual healthcare organizations and episodes of care. Rishel et al. (2005) provides a clear distinction and summary between a CPR, EMR and EHR. The CPR system provides support for all of the activities and processes involved in the delivery of clinical care, while the EMR is a CPR that is optimized to support ambulatory settings.

The EHR is the “system that integrates the other systems” and comprises of all the interconnected CRPs, EMRs and so on. Rishel et al. (2005) points out that it is common in the United States to view the EHR as being equivalent to the EMR. This is clearly an error and should be identified as such when it happens.

12.3. Longitudinal Patient Record and Decision Making

According to Green and Bowie (2004:95), the electronic health record (EHR) or a computer-based patient record (CPR) best facilitate the creation of a **longitudinal patient record**. The longitudinal patient record (LPR) contains records from different episodes of care, providers and facilities that are linked to form a view, over time, of a patient health care encounter (Green and Bowie, 2004:95). The longitudinal patient record facilitates clinical decision support, analysis of diagnoses and treatments and best practice multidisciplinary guidelines (Green and Bowie, 2004:95). Van Bemmel and Mussen (1999) define a longitudinal patient record as a patient’s record that contains data covering a period longer than one diseases episode.

Green and Bowie (2004:95) strongly argue that organizations move away from *paper-based* longitudinal patient records. Several key reasons were pointed out and are as follows: health records are not standardized, they sometimes are missing data and they are difficult to organization into a view of patient’s health over time. Instead, the authors recommend that organizations implement a clinical data repository, which allows for the collection of all clinical data in on centralized database. This would in turn provide easy access to data in electronic form to the patient’s clinical history.

DeGruy (2000) suggested that since a CPR store a large quantity of patient data on test results, medications, prior diagnoses, and other medical history, it could be a valuable source of information if interrogated with Knowledge Discovery in Databases (KDD) techniques. DeGruy (2000) argues that organizations with data warehouses may have shortened KDD implementation periods because their data has already been somewhat scrubbed and cleaned. Clean data can be a competitive advantage when leveraged properly, and KDD is the tool to accomplish that. Several examples of KDD include identifying patients who should receive flu shots, identifying patients who should enroll in a disease management program and ***identifying patients who are not in compliance with a treatment plan***.

According to Berndt et al. (2003:374) transaction-based star schemas can provide very useful functionality within a data warehouse framework. This means that all the different star schemas will form important components for constructing the LPR from the FSDOH data warehouse.

12.4. Implementing the Mapping Table

The proposed mapping table was implemented by deriving fields from the GRLS_ARV_PATIENTS_LINKED table. This implemented mapping table was called PATIENTS_MAPPED_DIM. The <datamart>_GRLS_NEWID columns were added for each data mart respectively. Thereafter the <datamart>_GRLS_NEWID columns were linked to the GRLS two-file linkage results between ARV and each data mart as was described in detail in the previous chapter. Finally the patient dimension table of each data mart was linked to the GRLS two-file linkage table for the data mart. The implemented process is depicted below (see figure 12-1) and provided the FSDOH warehouse with a conformed patient dimension table.

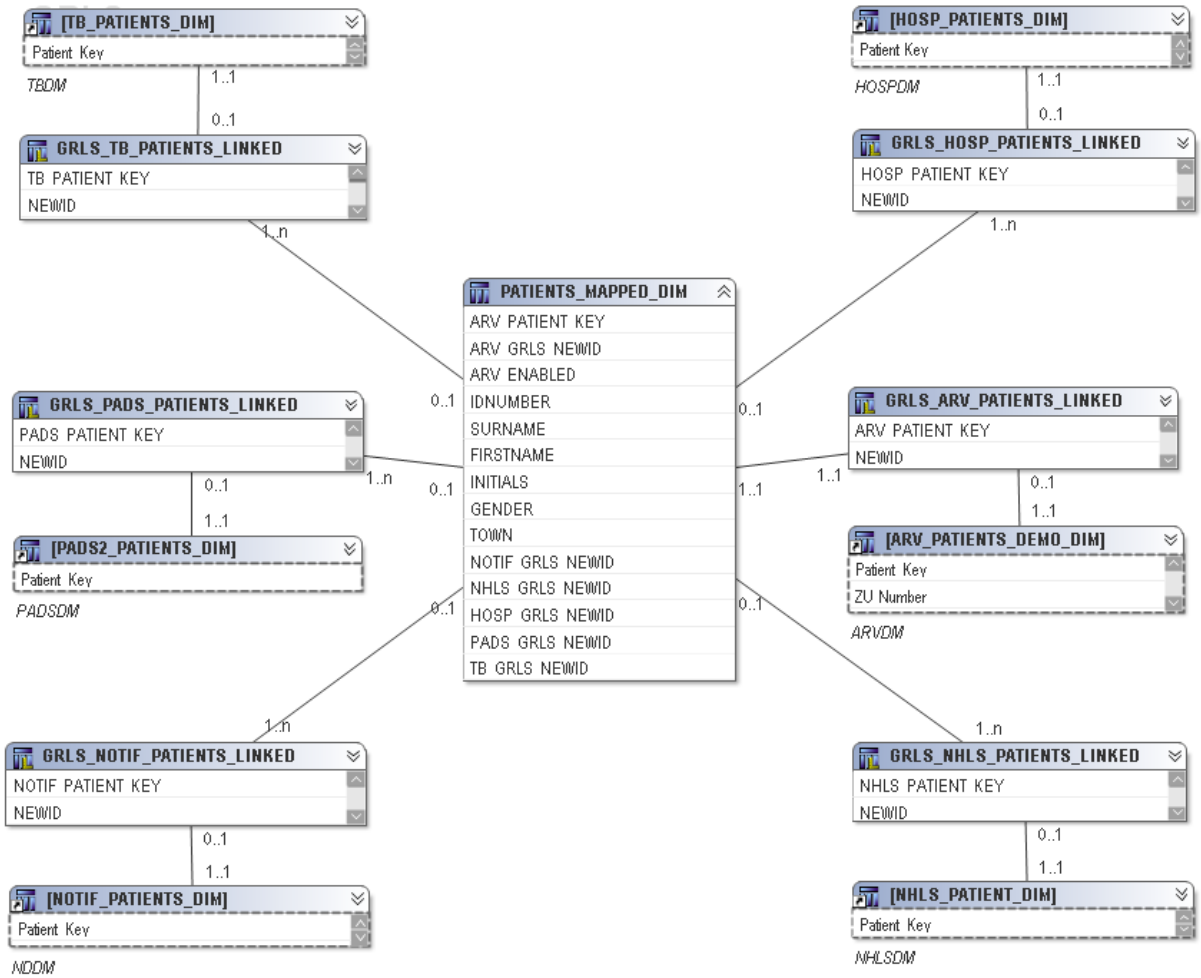


Figure 12-1: Implemented mapping table

12.5. Constructing the Longitudinal Patient Record

With the mapping table and related dimension tables now linked together the next step was to develop an LPR algorithm. The algorithm was developed in Oracle 10 release 2 using native SQL and PL/SQL. Depicted below (see figure 12-2) is the base LPR algorithm.

```
DECLARE

  i_zunumber              VARCHAR2(20) := 'ZU00003450';
  i_arv_grls_newid        NUMBER := 0;
  i_counter               NUMBER := 0;

BEGIN

  SELECT DISTINCT ARV_GRLS_NEWID  INTO i_arv_grls_newid
  FROM PROJECT1.PATIENTS_MAPPED_DIM
  WHERE ARV_PATIENT_KEY = i_zunumber;

  /* MPM ARV */

  FOR a IN (SELECT ARV_PATIENT_KEY, ARV_GRLS_NEWID
              FROM PROJECT1.PATIENTS_MAPPED_DIM
              WHERE ARV_GRLS_NEWID=i_arv_grls_newid) LOOP

    i_counter:=i_counter + 1;

    DBMS_OUTPUT.PUT_LINE ('*********************************************************************');
    DBMS_OUTPUT.PUT_LINE ('ARV Record # '||i_counter);
    DBMS_OUTPUT.PUT_LINE ('*********************************************************************');
    DBMS_OUTPUT.PUT_LINE ('.');
    DBMS_OUTPUT.PUT_LINE ('ZU          : '||a.ARV_PATIENT_KEY);
    DBMS_OUTPUT.PUT_LINE ('GRLS ID  : '||i_arv_grls_newid);
    DBMS_OUTPUT.PUT_LINE ('.');

  END LOOP;


  /* MPM ARV Events */


  FOR detail IN (SELECT NVL(COALESCE(a.ENTRY_DATE_KEY,
                                to_number(DATE_CAPTURED,'yyyymmdd')), '20040501') DATUM,
                                a.PATIENT_KEY,
                                b.ENTRY_TYPE_NAME,
                                c.TREATMENT_LOCATION_NAME,
                                a.DATA_SOURCE
                  FROM ARV_MEDICAL_RECORD_FACT a, ARV_TREATMENT_DIM b,
                          TREATMENT_LOCATION_DIM c
                  WHERE a.ENTRY_TYPE_KEY = b.ENTRY_TYPE_KEY
                  AND   (b.ENTRY_TYPE_KEY = 100
                    OR    b.ENTRY_TYPE_KEY BETWEEN 900 AND 907
                    OR    b.ENTRY_TYPE_KEY BETWEEN 200 AND 212)
                  AND   a.TREATMENT_LOCATION_KEY = c.TREATMENT_LOCATION_KEY
                  AND   a.PATIENT_KEY IN (SELECT ARV_PATIENT_KEY
                                          FROM PROJECT1.PATIENTS_MAPPED_DIM
                                          WHERE ARV_GRLS_NEWID=i_arv_grls_newid)
                  ORDER BY DATUM) LOOP

    /* output results here */

  END LOOP;
```

```
/* Notifiable Diseases Events */


 FOR master1 IN (SELECT NOTIF_GRLS_NEWID
                      FROM project1.PATIENTS_MAPPED_DIM
                      WHERE ARV_GRLS_NEWID = i_arv_grls_newid) LOOP

  FOR master2 IN (SELECT NOTIF_PATIENT_KEY
                       FROM project1.GRLS_NOTIF_PATIENTS_LINKED
                       WHERE NEWID = master1.NOTIF_GRLS_NEWID) LOOP

    FOR detail IN (SELECT a.IDENTIFIED_DATE_KEY,
                                b.DISEASE_NAME,
                                c.TREATMENT_LOCATION_NAME
                      FROM NOTIF_FACT a, NOTIF_DISEASE_DIM b, TREATMENT_LOCATION_DIM c
                      WHERE a.DISEASE_KEY = b.DISEASE_KEY
                         AND   a.TREATMENT_LOCATION_KEY = c.TREATMENT_LOCATION_KEY
                         AND   a. PATIENT_KEY IN master2.NOTIF_PATIENT_KEY) LOOP

     /* output results here */

    END LOOP;

  END LOOP;

END LOOP;


/* NHLS Bloodresults Events */


FOR master1 IN (SELECT NHLS_GRLS_NEWID
                      FROM project1.PATIENTS_MAPPED_DIM
                      WHERE ARV_GRLS_NEWID = i_arv_grls_newid) LOOP

  FOR master2 IN (SELECT NHLS_PATIENT_KEY
                       FROM project1.GRLS_NHLS_PATIENTS_LINKED
                       WHERE NEWID = master1.NHLS_GRLS_NEWID) LOOP

    FOR detail IN (SELECT a.TEST_DATE_KEY,
                                a.TEST_NUMBER,
                                b.TEST_TYPE,
                                c.TREATMENT_LOCATION_NAME,
                                a.CD4_ABSOLUTE_COUNT,
                                a.VIRAL_LOAD
                      FROM NHLS_FACT a, NHLS_TESTS_DIM b, TREATMENT_LOCATION_DIM c
                      WHERE a.TEST_KEY = b.TEST_KEY
                         AND   a.TREATMENT_LOCATION_KEY = c.TREATMENT_LOCATION_KEY
                         AND   a.PATIENT_KEY IN master2.NHLS_PATIENT_KEY
                      ORDER BY a.TEST_DATE_KEY) LOOP

      /* output results here */

     END LOOP;

   END LOOP;

END LOOP;
```

```
 /* MPM Bloodresults Events */


  FOR detail IN (SELECT COALESCE(a.ENTRY_DATE_KEY,to_number(DATE_CAPTURED,'yyyymmdd'))
DATUM,
                              a.PATIENT_KEY,
                              b.ENTRY_TYPE_NAME,
                              c.TREATMENT_LOCATION_NAME,
                              a.DATA_SOURCE,
                              a.FACT_VALUE
                    FROM ARV_MEDICAL_RECORD_FACT a, ARV_TREATMENT_DIM b,
                           TREATMENT_LOCATION_DIM c
                    WHERE a.ENTRY_TYPE_KEY = b.ENTRY_TYPE_KEY
                       AND   (b.ENTRY_TYPE_KEY BETWEEN 300 AND 301
                        OR    b.ENTRY_TYPE_KEY BETWEEN 303 AND 318)
                      AND   a.TREATMENT_LOCATION_KEY = c.TREATMENT_LOCATION_KEY
                      AND   a.PATIENT_KEY IN (SELECT ARV_PATIENT_KEY
                                              FROM PROJECT1.PATIENTS_MAPPED_DIM
                                              WHERE ARV_GRLS_NEWID=i_arv_grls_newid)
                    ORDER BY DATUM) LOOP

     /* output results here */

  END LOOP;


/* MPM Regimen */


  FOR detail IN (SELECT COALESCE(a.ENTRY_DATE_KEY,
                        to_number(DATE_CAPTURED,'yyyymmdd')) DATUM,
                        a.PATIENT_KEY,
                        b.ENTRY_TYPE_NAME,
                        c.TREATMENT_LOCATION_NAME,
                        a.DATA_SOURCE,
                        a.FORM_SOURCE,
                        a.FACT_VALUE,
                        d.REGIMEN_KEY,
                        d.REGIMEN_SEQUENCE
                   FROM ARV_MEDICAL_RECORD_FACT a, ARV_TREATMENT_DIM b,
                     TREATMENT_LOCATION_DIM c,
                     ARV_REGIMEN_DIM d
                   WHERE a.ENTRY_TYPE_KEY = b.ENTRY_TYPE_KEY
                      AND   b.ENTRY_TYPE_KEY = 400
                      AND   a.TREATMENT_LOCATION_KEY = c.TREATMENT_LOCATION_KEY
                      AND   a.FACT_STRING  = d.REGIMEN_KEY
                      AND   a.PATIENT_KEY IN (SELECT ARV_PATIENT_KEY
                                              FROM PROJECT1.PATIENTS_MAPPED_DIM
                                              WHERE ARV_GRLS_NEWID=i_arv_grls_newid)
             ORDER BY DATUM) LOOP

     i_counter:=i_counter + 1;

     DBMS_OUTPUT.PUT_LINE ('ARV Regimen: '||i_counter||' for '||detail.PATIENT_KEY);
     DBMS_OUTPUT.PUT_LINE ('ARV Regimen Date              : '||detail.DATUM);
     DBMS_OUTPUT.PUT_LINE ('ARV Regimen Captured On   : '||detail.FORM_SOURCE);
     DBMS_OUTPUT.PUT_LINE ('ARV Regimen                     : '||to_char(detail.REGIMEN_KEY));

     drugstring := '';
```

```
  IF to_char(detail.REGIMEN_KEY) NOT IN ('0','1','2') THEN
    IF (substr(detail.REGIMEN_SEQUENCE,1,1)='Y') THEN drugstring := drugstring || '3TC '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,2,1)='Y') THEN drugstring := drugstring || 'D4T '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,3,1)='Y') THEN drugstring := drugstring || 'EFV '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,4,1)='Y') THEN drugstring := drugstring || 'NVP '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,5,1)='Y') THEN drugstring := drugstring || 'AZT '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,6,1)='Y') THEN drugstring := drugstring || 'DDI '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,7,1)='Y') THEN drugstring := drugstring || 'LPV 400/100 '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,8,1)='Y') THEN drugstring := drugstring || 'LPV 100/100 '; END IF;
    DBMS_OUTPUT.PUT_LINE ('ARV Regimen Drugs (adult) : '||drugstring);
    DBMS_OUTPUT.PUT_LINE ('.');
  ELSE
    IF (substr(detail.REGIMEN_SEQUENCE,1,1)='Y') THEN drugstring := drugstring || '3TC '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,2,1)='Y') THEN drugstring := drugstring || 'D4T '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,3,1)='Y') THEN drugstring := drugstring || 'EFV '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,4,1)='Y') THEN drugstring := drugstring || 'NVP '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,5,1)='Y') THEN drugstring := drugstring || 'AZT '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,6,1)='Y') THEN drugstring := drugstring || 'ABC '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,7,1)='Y') THEN drugstring := drugstring || 'RITO '; END IF;
    IF (substr(detail.REGIMEN_SEQUENCE,8,1)='Y') THEN drugstring := drugstring || 'RITOLOP '; END IF;
    DBMS_OUTPUT.PUT_LINE ('ARV Regimen Drugs (children) : '||drugstring);
    DBMS_OUTPUT.PUT_LINE ('.');
  END IF;

END LOOP;



  /* MPM Weight */



  i_counter := 0;

  FOR detail IN (SELECT COALESCE(a.ENTRY_DATE_KEY,
                        to_number(DATE_CAPTURED,'yyyymmdd')) DATUM,
                        a.PATIENT_KEY,
                        b.ENTRY_TYPE_NAME,
                        c.TREATMENT_LOCATION_NAME,
                        a.DATA_SOURCE,
                        a.FORM_SOURCE,
                        a.FACT_VALUE
                 FROM ARV_MEDICAL_RECORD_FACT a, ARV_TREATMENT_DIM b,
                      TREATMENT_LOCATION_DIM c
                 WHERE a.ENTRY_TYPE_KEY = b.ENTRY_TYPE_KEY
                   AND   b.ENTRY_TYPE_KEY = 302
                   AND   a.TREATMENT_LOCATION_KEY = c.TREATMENT_LOCATION_KEY
                   AND   a.PATIENT_KEY IN (SELECT ARV_PATIENT_KEY
                                           FROM PROJECT1.PATIENTS_MAPPED_DIM
                                           WHERE ARV_GRLS_NEWID=i_arv_grls_newid)
                 ORDER BY DATUM) LOOP

     i_counter:=i_counter + 1;

     DBMS_OUTPUT.PUT_LINE ('ARV Weight: '||i_counter||' for '||detail.PATIENT_KEY);
     DBMS_OUTPUT.PUT_LINE ('ARV Weight Date             : '||detail.DATUM);
     DBMS_OUTPUT.PUT_LINE ('ARV Weight Captured On  : '||detail.FORM_SOURCE);
     DBMS_OUTPUT.PUT_LINE ('ARV Weight (kg)             : '||detail.FACT_VALUE);
     DBMS_OUTPUT.PUT_LINE ('.');

  END LOOP;
```

```
/* MEDITECH Events */

 FOR master1 IN (SELECT HOSP_GRLS_NEWID
                      FROM project1.PATIENTS_MAPPED_DIM
                      WHERE ARV_GRLS_NEWID = i_arv_grls_newid) LOOP

   FOR master2 IN (SELECT HOSP_PATIENT_KEY
                       FROM project1.GRLS_HOSP_PATIENTS_LINKED
                       WHERE NEWID = master1.HOSP_GRLS_NEWID) LOOP

     FOR detail IN (SELECT a.ADMISSION_DATE_KEY,
                           a.DISCHARGE_DATE_KEY,
                           b.VISIT_TYPE,
                           b.VISIT_REASON,
                           c.TREATMENT_LOCATION_NAME,
                           d.ICD10_CODE,
                           d.ICD10_DESCRIPTION
                      FROM HOSP_MEDICAL_RECORD_FACT a, HOSP_VISIT_DIM b,
                           TREATMENT_LOCATION_DIM c,
                           HOSP_ICD10_DIM d
                      WHERE a.VISIT_KEY = b.VISIT_KEY
                        AND   a.ICD10_KEY = d.ICD10_KEY
                        AND   a.TREATMENT_LOCATION_KEY = c.TREATMENT_LOCATION_KEY
                        AND   PATIENT_KEY IN master2.HOSP_PATIENT_KEY
                      ORDER BY a.ADMISSION_DATE_KEY) LOOP

       /* output results here */

     END LOOP;

   END LOOP;

 END LOOP;

/* PADS Events */

 FOR master1 IN (SELECT PADS_GRLS_NEWID
                      FROM project1.PATIENTS_MAPPED_DIM
                      WHERE ARV_GRLS_NEWID = i_arv_grls_newid) LOOP

   FOR master2 IN (SELECT PADS_PATIENT_KEY
                       FROM project1.GRLS_PADS_PATIENTS_LINKED
                       WHERE NEWID = master1.PADS_GRLS_NEWID) LOOP

     FOR detail IN (SELECT a.VISIT_ADMISSION_DATE_KEY,
                           a.VISIT_DISCHARGE_DATE_KEY,
                           a.VISIT_TYPE,
                           a.DISCHARGE_DISPOSITION,
                           a.DISCHARGE_FINAL_DIAGNOSIS,
                           a.DISCHARGE_ICD10_1,
                           a.PAT_SURNAME,
                           a.PAT_FIRSTNAME,
                           a.PAT_INITIALS,
                           c.TREATMENT_LOCATION_NAME
                      FROM wh_pads2.VISIT_DETAILS a, TREATMENT_LOCATION_DIM c
                      WHERE  a.FACILITY_KEY = c.TREATMENT_LOCATION_KEY
                        AND   PATIENT_KEY IN master2.PADS_PATIENT_KEY
                      ORDER BY a.VISIT_ADMISSION_DATE_KEY) LOOP

       /* output results here */

     END LOOP;

   END LOOP;

 END LOOP;
```

```
/* TB Events */


FOR master1 IN (SELECT TB_GRLS_NEWID
                    FROM project1.PATIENTS_MAPPED_DIM
                    WHERE ARV_GRLS_NEWID = i_arv_grls_newid) LOOP

  FOR master2 IN (SELECT TB_PATIENT_KEY
                     FROM project1.GRLS_TB_PATIENTS_LINKED
                     WHERE NEWID = master1.TB_GRLS_NEWID) LOOP

    FOR detail IN (SELECT a.REGISTRATION_DATE_KEY,
                          b.TB_TRANSFER,
                          b.TB_REGIMEN,
                          b.TB_CATEGORY,
                          b.TB_CLASSIFICATION,
                          c.TREATMENT_LOCATION_NAME
                    FROM TB_FACT a, TB_JUNK_DIM b, TREATMENT_LOCATION_DIM c
                    WHERE a.JUNK_KEY = b.JUNK_KEY
                       AND   a.TREATMENT_LOCATION_KEY = c.TREATMENT_LOCATION_KEY
                       AND   PATIENT_KEY IN master2.TB_PATIENT_KEY
                    ORDER BY a.REGISTRATION_DATE_KEY) LOOP

     /* output results here */

    END LOOP;

  END LOOP;

 END LOOP;


/* Home Affairs */


 FOR master1 IN (SELECT MAX(IDNUMBER) IDNUMBER
                     FROM project1.PATIENTS_MAPPED_DIM
                     WHERE ARV_GRLS_NEWID = i_arv_grls_newid) LOOP

   FOR detail IN (SELECT nvl(a.ID_NUMBER_STATUS,'ALIVE') ID_NUMBER_STATUS,
                         to_char(a.DATE_OF_DEATH,'yyyymmdd') DATE_OF_DEATH,
                         a.PLACE_OF_DEATH,
                         a.CAUSE_OF_DEATH
                  FROM DEATHREGMATCH_FINAL a
                  WHERE a.ID_NUMBER IN master1.IDNUMBER
                  ORDER BY a.DATE_OF_DEATH) LOOP

      DBMS_OUTPUT.PUT_LINE ('Person status       : '||detail.ID_NUMBER_STATUS);
      DBMS_OUTPUT.PUT_LINE ('Date of death       : '||detail.DATE_OF_DEATH);
      DBMS_OUTPUT.PUT_LINE ('Place of death      : '||detail.PLACE_OF_DEATH);
      DBMS_OUTPUT.PUT_LINE ('Cause of death      : '||detail.CAUSE_OF_DEATH);
      DBMS_OUTPUT.PUT_LINE ('.');

   END LOOP;

 END LOOP;

END;
```

Figure 12-2: Base LPR algorithm

12.6. Interfacing with the Longitudinal Patient Record

A decision was taken to develop a web based application on the existing Oracle Portal platform that is being used in the FSDOH. Oracle PL/SQL and Oracle Web Toolkit were used for this development to ensure a low-bandwidth solution that can be accessible by any user. Access to the web based LPR was established by requesting a URL from within the Meditech MPM application (see figure 12-3) when the user selects the Link LPR button. The parameters Username and the Patient's key were passed to the LPR web application. Section 12.7 will outline the process of authentication and authorization and how these passed parameters were used.

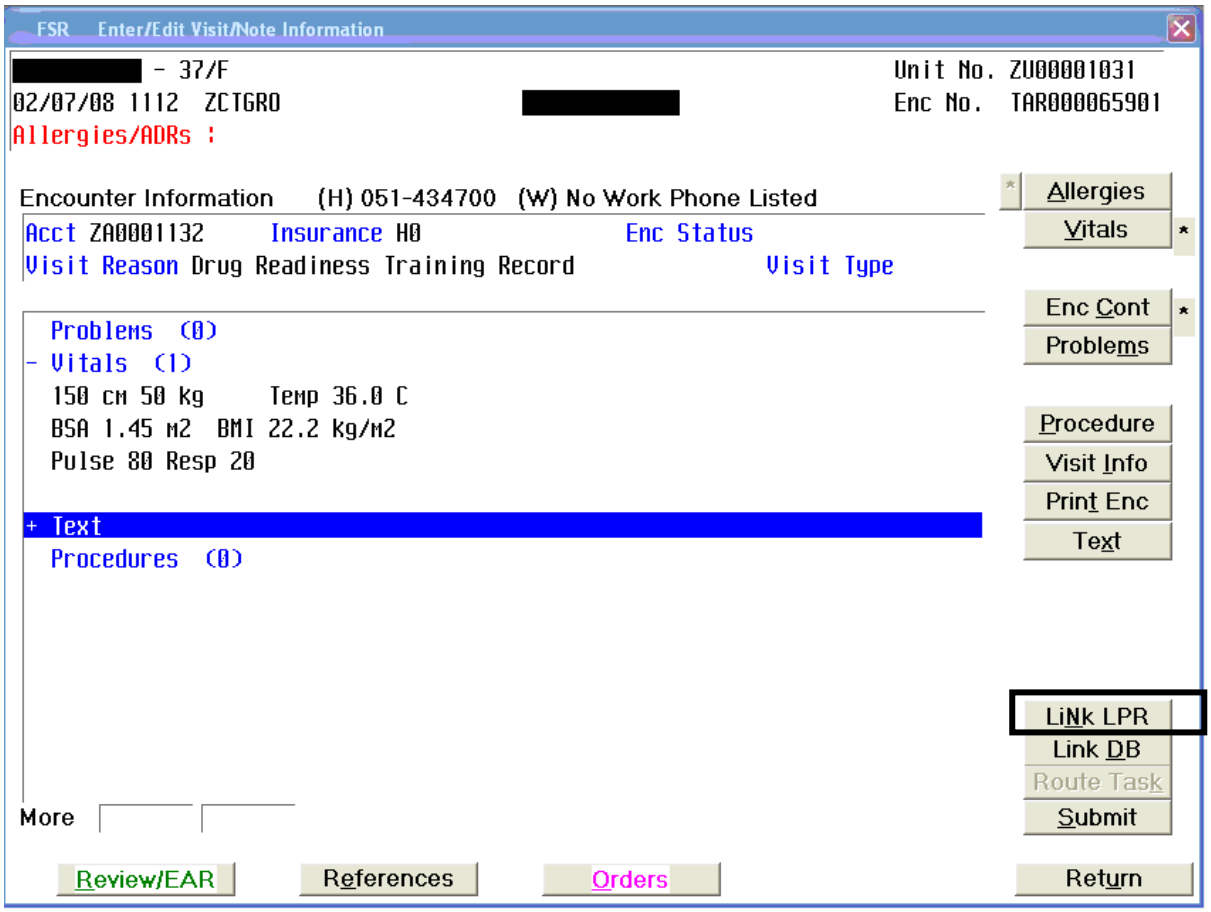


Figure 12-3: Screenshot of the LPR linkage button in MPM

12.7. Securing the Longitudinal Patient Record

Patient confidentiality is an important issue when dealing with sensitive topics such as HIV and AIDS. In South Africa the issue is further complicated with the fact that it is not a notifiable disease and therefore the HIV status of a patient is confidential. In an attempt to secure the data, additional security measures were implemented by altering the original ARV_USERS_DIM table (see figure 12-4). The ARV_USERS_DIM table contained all the Meditech MPM user ids and usernames. Two columns were added namely PERSAL_JOB_TITLE and PERSAL_OCCUPATIONAL_CLASS (see figure 6-10 for the ARVDM dimensional model). Both the columns are updated from the HRDM by matching the PERSAL numbers. If the user requesting the LPR was of the occupational class NURSE or DOCTER, he or she had to enter their PERSAL number as password. See figure 12-5 for the user authorization screen. For any other user, the screen will be blank with a message saying "Unauthorized Access".

COLUMN ID	Column Name	Data Type	Nullable
1	MTUSERNAME	VARCHAR2(50 BYTE)	No
2	USERNAME	VARCHAR2(100 BYTE)	Yes
3	PERSAL	VARCHAR2(100 BYTE)	Yes
4	PERSAL_JOB_TITLE	VARCHAR2(50 BYTE)	Yes
5	PERSAL_OCCUPATIONAL_CLASS	VARCHAR2(50 BYTE)	Yes

Figure 12-4: Modified ARV_USERS_DIM table

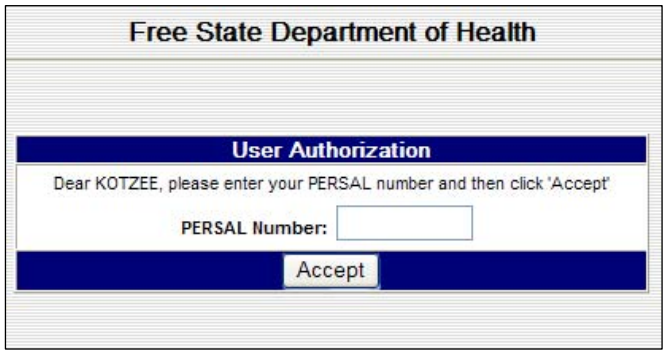


Figure 12-5: Screenshot of user authentication

When the user has successfully authenticated with the correct PERSAL number, and is also of the class NURSE or DOCTOR the LPR will be displayed for a patient. The following section will provide a step-by-step explanation of how the LPR was used to provide an integrated view of a patient.

12.8. Example of a Longitudinal Patient Record

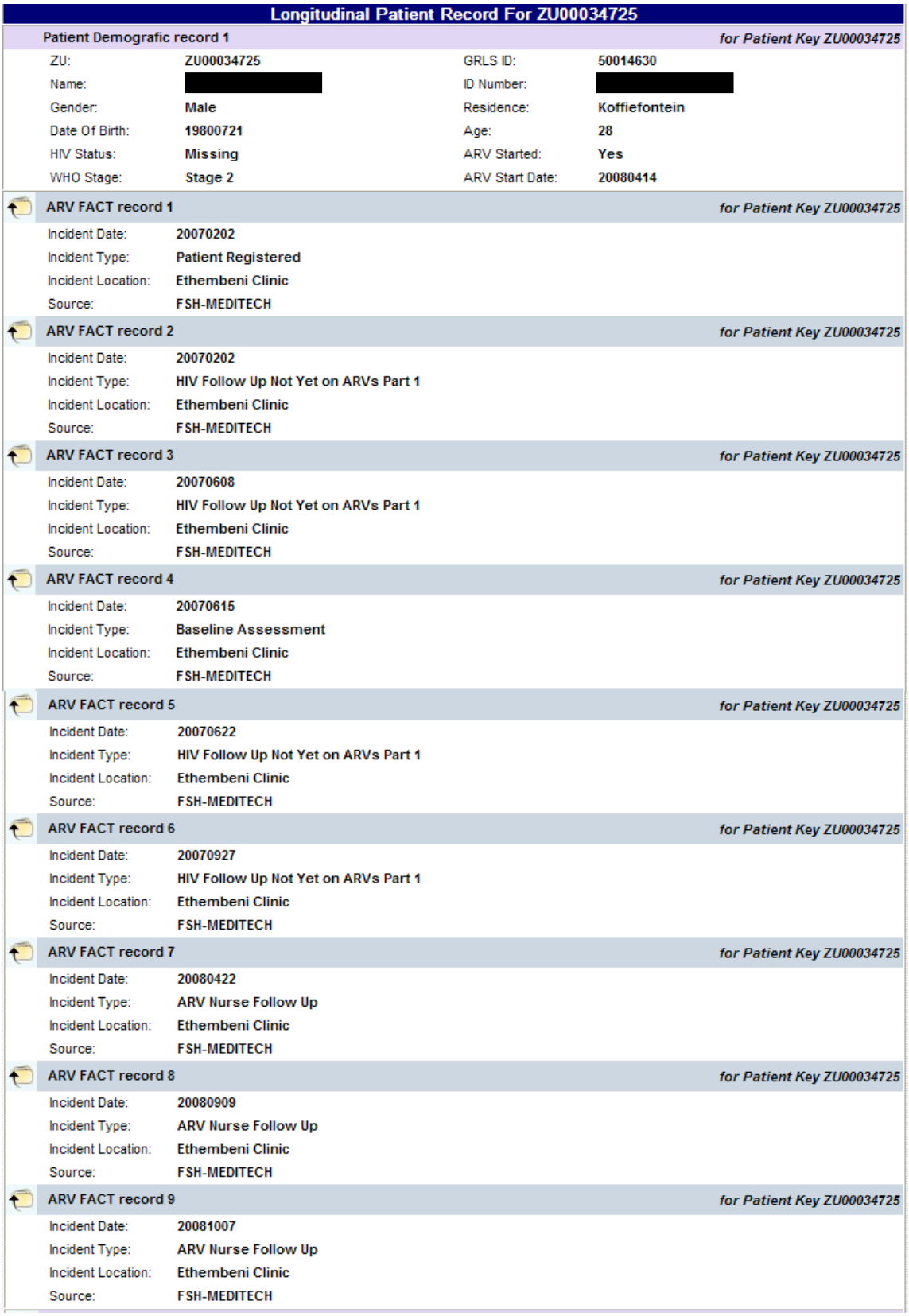
The following two sections will describe examples using the LPR. The first example will be a patient having a single EPR on the MPM system. The second example will be a patient with duplicate EPRs on the MPM system.

12.8.1. Single Longitudinal Patient Record

For the following discussion a registered patient will be used as an example to illustrate the value of the LPR. The example patient was registered on the ARV programme in February 2007. Several ARV treatment episodes have been recorded for the patient in the MPM system at Ethembeni Clinic (Koffiefontein). By examining the incident history, a good ARV treatment adherence rate can be observed. CD4 and Viral Load (VL) blood test results were received from NHLS that were recorded at Ethembeni Clinic (Koffiefontein). These blood test results confirmed that the patient was eligible for treatment (CD4 < 200) and should be receiving antiretroviral therapy. Several in-patient episodes (diagnosed with bacterial meningitis) were recorded on the Meditech HIS system and occurred at the Pelonomi Hospital (Bloemfontein). One other in-patient visit was recorded on the PADS system and occurred at the Diamant Hospital (Jagersfontein). A TB episode was also discovered for the patient on the TB register that was recorded at Ethembeni Clinic (Koffiefontein). According to the Department of Home Affairs the patient was also still alive when this example was compiled. By placing these incidents on a map (see figure 12-6) using a black pin, one can visualize the movement of the patient between points of service, each using a different information system. Each of these information systems were converted into a data mart in the FSDOH data warehouse.

Figure 12-6: Map of the Free State with LPR example

The following series of screenshots (see figures 12-7 to 12-11) will be based on the previous patient using the web based LPR application. It will demonstrate the implementation of the linkage exercise and how all the incidents where combined into a single easy-to-use screen for a clinician or nurse.



Longitudinal Patient Record For ZU00034725

Patient Demografic record 1 *for Patient Key ZU00034725*

ZU:	ZU00034725	GRLS ID:	50014630
Name:	[redacted]	ID Number:	[redacted]
Gender:	Male	Residence:	Koffiefontein
Date Of Birth:	19800721	Age:	28
HIV Status:	Missing	ARV Started:	Yes
WHO Stage:	Stage 2	ARV Start Date:	20080414

ARV FACT record 1 *for Patient Key ZU00034725*

Incident Date:	20070202
Incident Type:	Patient Registered
Incident Location:	Ethembeni Clinic
Source:	FSH-MEDITECH

ARV FACT record 2 *for Patient Key ZU00034725*

Incident Date:	20070202
Incident Type:	HIV Follow Up Not Yet on ARVs Part 1
Incident Location:	Ethembeni Clinic
Source:	FSH-MEDITECH

ARV FACT record 3 *for Patient Key ZU00034725*

Incident Date:	20070608
Incident Type:	HIV Follow Up Not Yet on ARVs Part 1
Incident Location:	Ethembeni Clinic
Source:	FSH-MEDITECH

ARV FACT record 4 *for Patient Key ZU00034725*

Incident Date:	20070615
Incident Type:	Baseline Assessment
Incident Location:	Ethembeni Clinic
Source:	FSH-MEDITECH

ARV FACT record 5 *for Patient Key ZU00034725*

Incident Date:	20070622
Incident Type:	HIV Follow Up Not Yet on ARVs Part 1
Incident Location:	Ethembeni Clinic
Source:	FSH-MEDITECH

ARV FACT record 6 *for Patient Key ZU00034725*

Incident Date:	20070927
Incident Type:	HIV Follow Up Not Yet on ARVs Part 1
Incident Location:	Ethembeni Clinic
Source:	FSH-MEDITECH

ARV FACT record 7 *for Patient Key ZU00034725*

Incident Date:	20080422
Incident Type:	ARV Nurse Follow Up
Incident Location:	Ethembeni Clinic
Source:	FSH-MEDITECH

ARV FACT record 8 *for Patient Key ZU00034725*

Incident Date:	20080909
Incident Type:	ARV Nurse Follow Up
Incident Location:	Ethembeni Clinic
Source:	FSH-MEDITECH

ARV FACT record 9 *for Patient Key ZU00034725*

Incident Date:	20081007
Incident Type:	ARV Nurse Follow Up
Incident Location:	Ethembeni Clinic
Source:	FSH-MEDITECH

Figure 12-7: ARV Incidents from the LPR

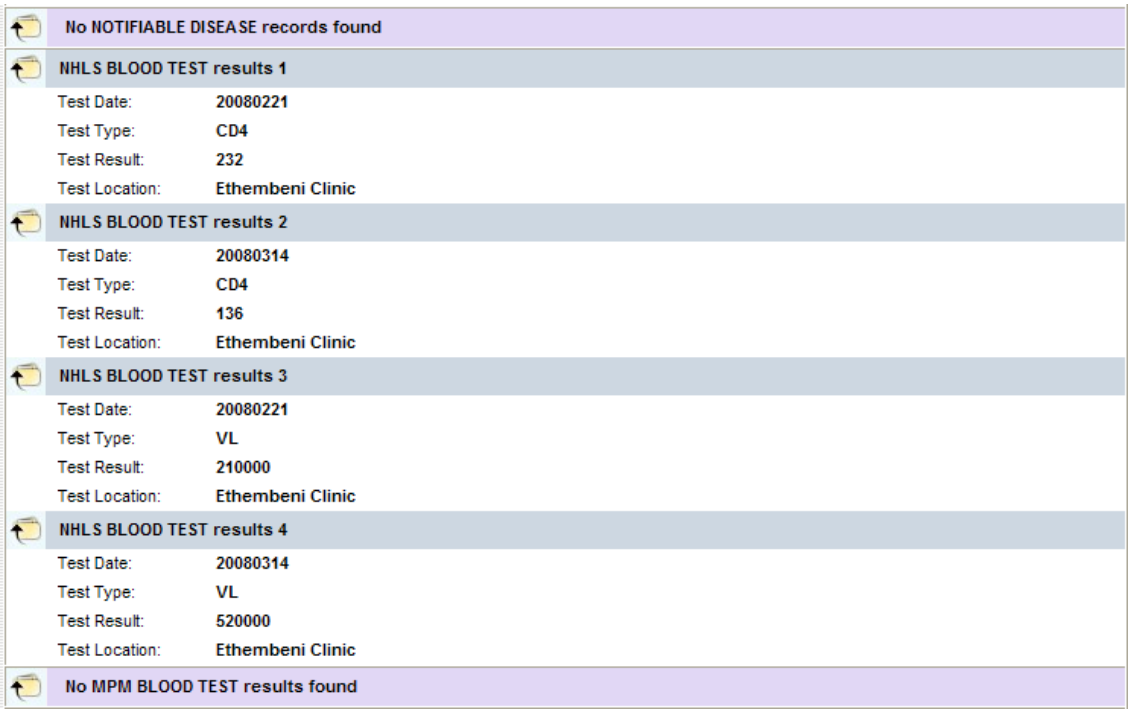


Figure 12-8: Notifiable Diseases, NHLS and MPM Blood Results from the LPR

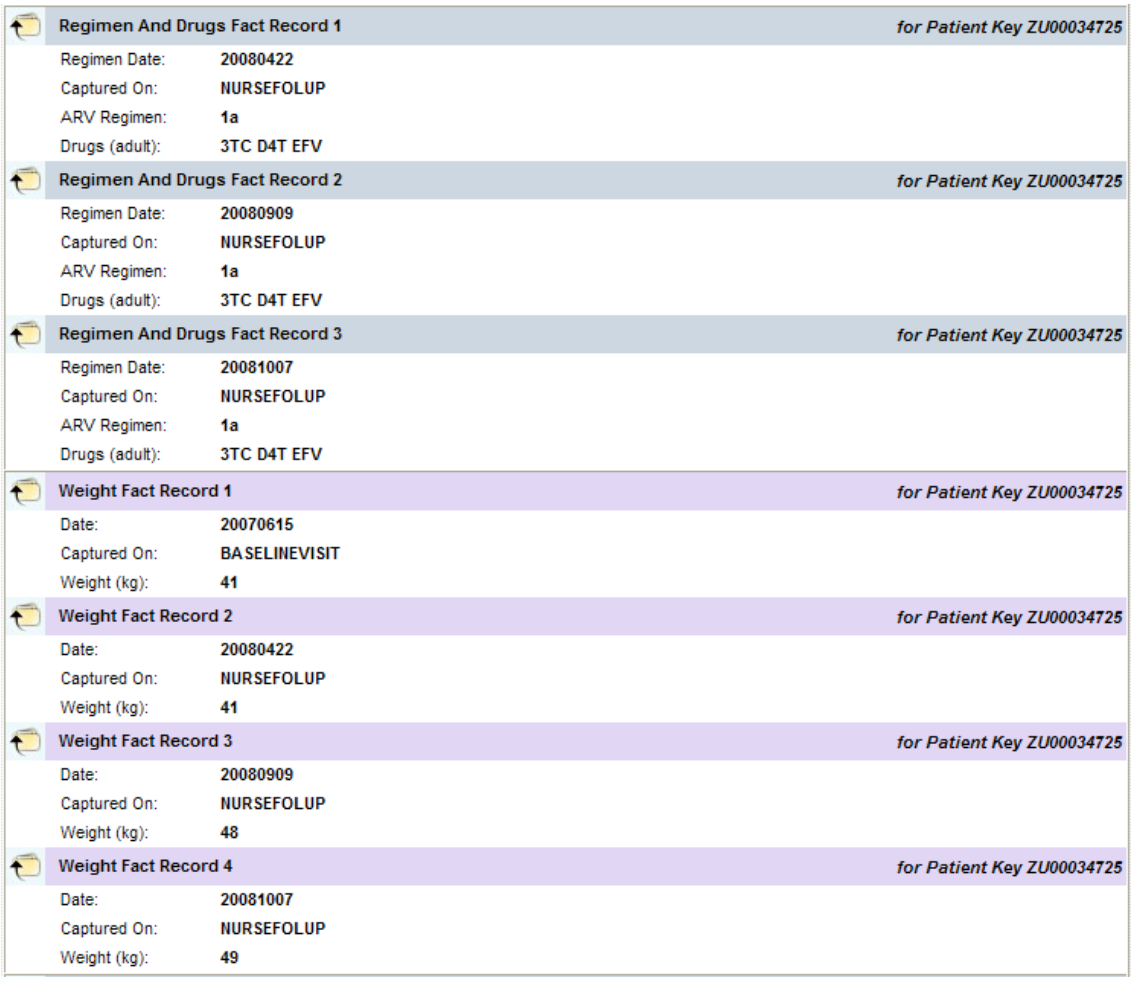
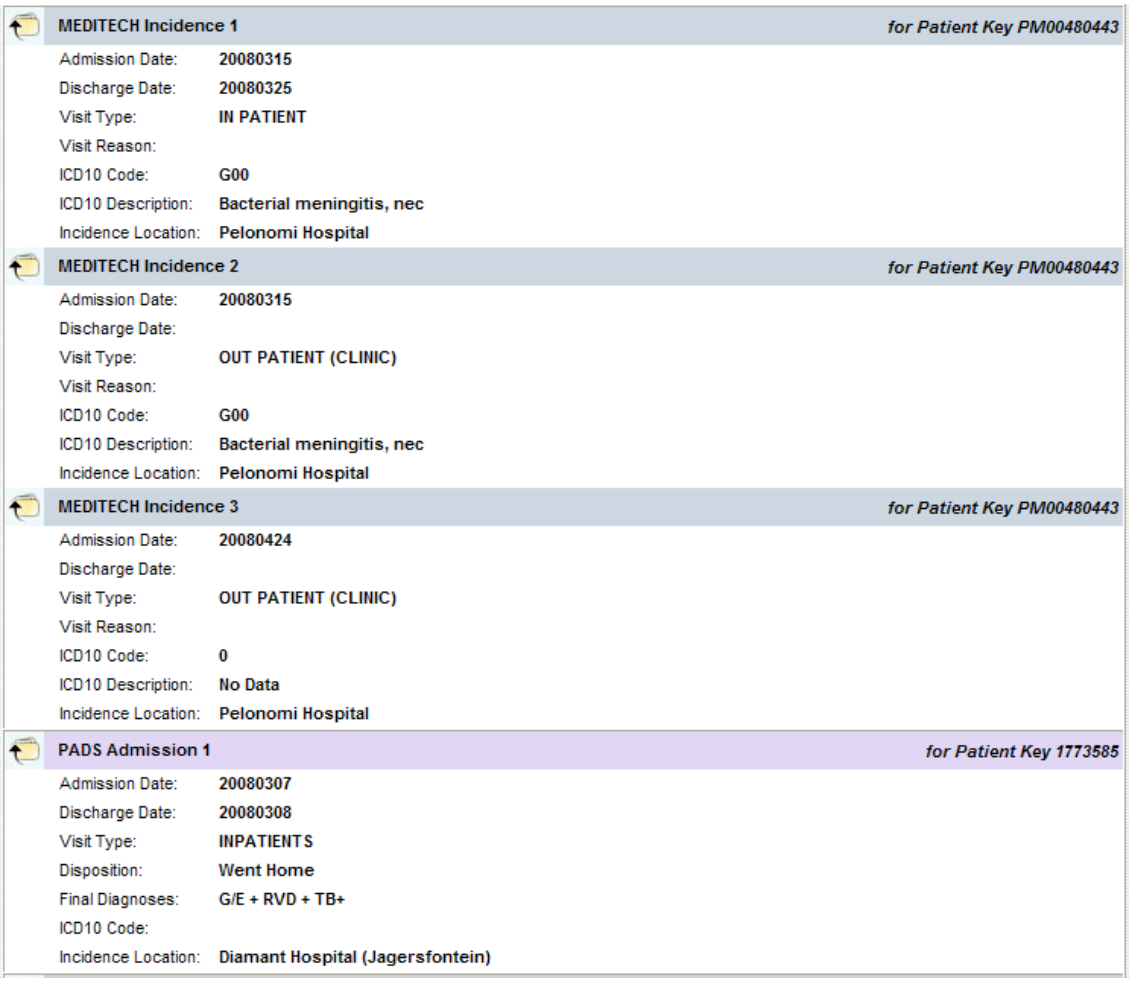


Figure 12:9: Regimen, Drugs and Weight Records from the LPR



MEDITECH Incidence 1 *for Patient Key PM00480443*

Admission Date:	20080315
Discharge Date:	20080325
Visit Type:	IN PATIENT
Visit Reason:	
ICD10 Code:	G00
ICD10 Description:	Bacterial meningitis, nec
Incidence Location:	Pelonomi Hospital

MEDITECH Incidence 2 *for Patient Key PM00480443*

Admission Date:	20080315
Discharge Date:	
Visit Type:	OUT PATIENT (CLINIC)
Visit Reason:	
ICD10 Code:	G00
ICD10 Description:	Bacterial meningitis, nec
Incidence Location:	Pelonomi Hospital

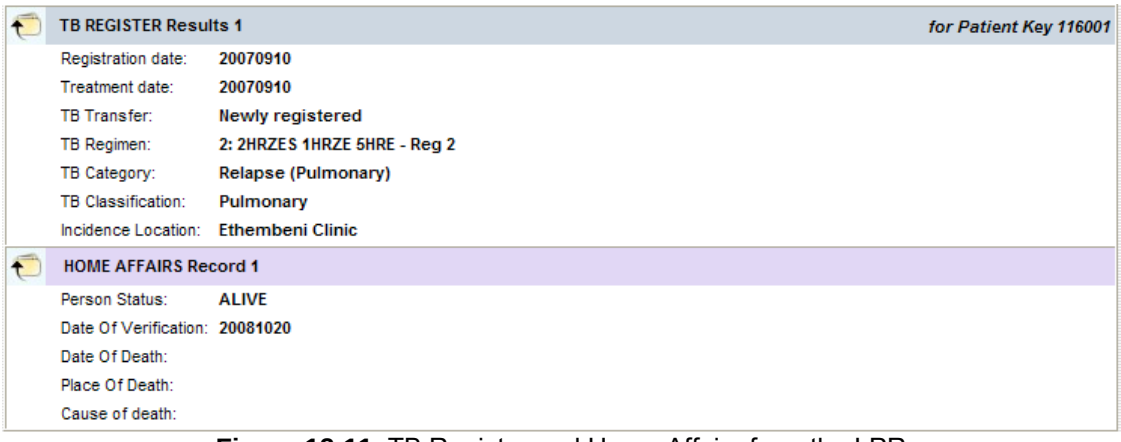
MEDITECH Incidence 3 *for Patient Key PM00480443*

Admission Date:	20080424
Discharge Date:	
Visit Type:	OUT PATIENT (CLINIC)
Visit Reason:	
ICD10 Code:	0
ICD10 Description:	No Data
Incidence Location:	Pelonomi Hospital

PADS Admission 1 *for Patient Key 1773585*

Admission Date:	20080307
Discharge Date:	20080308
Visit Type:	INPATIENTS
Disposition:	Went Home
Final Diagnoses:	G/E + RVD + TB+
ICD10 Code:	
Incidence Location:	Diamant Hospital (Jagersfontein)

Figure 12:10: Meditech and PADS Hospital Visits from the LPR



TB REGISTER Results 1 *for Patient Key 116001*

Registration date:	20070910
Treatment date:	20070910
TB Transfer:	Newly registered
TB Regimen:	2: 2HRZES 1HRZE 5HRE - Reg 2
TB Category:	Relapse (Pulmonary)
TB Classification:	Pulmonary
Incidence Location:	Ethembeni Clinic

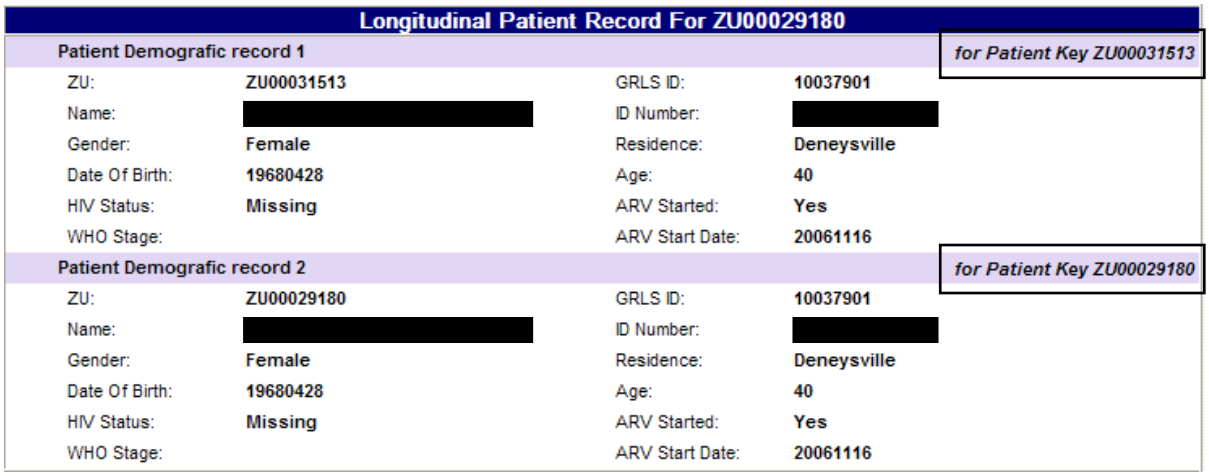
HOME AFFAIRS Record 1

Person Status:	ALIVE
Date Of Verification:	20081020
Date Of Death:	
Place Of Death:	
Cause of death:	

Figure 12:11: TB Register and Home Affairs from the LPR

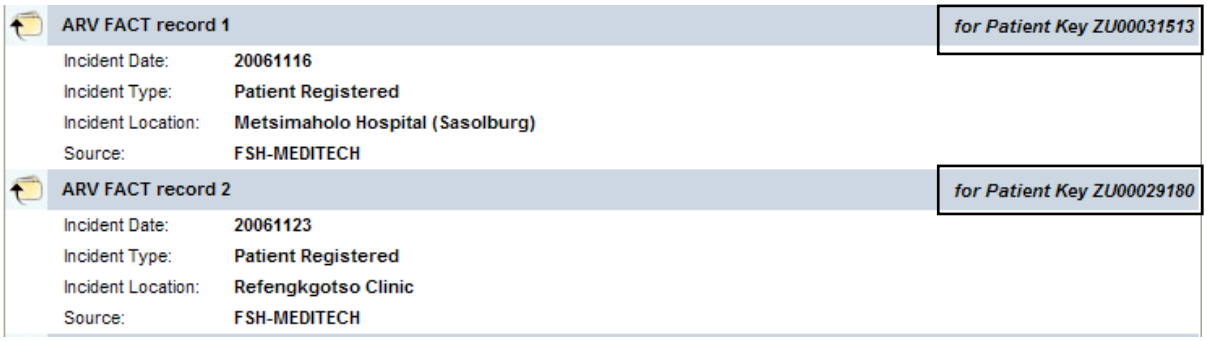
12.8.2. Duplicate Longitudinal Patient Record

For the following discussion a registered patient with **duplicate** EPR's will be used as an example to illustrate the value of the LPR. The example patient was registered on the ARV programme in November 2006. By using only OLTP query-based approaches the full picture of this patient would not have been possible. As can be seen from the example below, the patient has two different keys (ZU00031512 and ZU00029180), both with its own set of data. GRLS was able to link these two records together and provide a single key (GRLS ID 10037901) and the LPR used this single key to group all the data together (see figures 12-12 to 12-15).



Longitudinal Patient Record For ZU00029180				
Patient Demografic record 1				*for Patient Key ZU00031513*
ZU:	ZU00031513	GRLS ID:	10037901	
Name:	[redacted]	ID Number:	[redacted]	
Gender:	Female	Residence:	Deneysville	
Date Of Birth:	19680428	Age:	40	
HIV Status:	Missing	ARV Started:	Yes	
WHO Stage:		ARV Start Date:	20061116	
Patient Demografic record 2				*for Patient Key ZU00029180*
ZU:	ZU00029180	GRLS ID:	10037901	
Name:	[redacted]	ID Number:	[redacted]	
Gender:	Female	Residence:	Deneysville	
Date Of Birth:	19680428	Age:	40	
HIV Status:	Missing	ARV Started:	Yes	
WHO Stage:		ARV Start Date:	20061116	

Figure 12-12: Illustrating the grouping of the same patient's details



ARV FACT record 1		*for Patient Key ZU00031513*
Incident Date:	20061116	
Incident Type:	Patient Registered	
Incident Location:	Metsimaholo Hospital (Sasolburg)	
Source:	FSH-MEDITECH	
ARV FACT record 2		*for Patient Key ZU00029180*
Incident Date:	20061123	
Incident Type:	Patient Registered	
Incident Location:	Refengkgotso Clinic	
Source:	FSH-MEDITECH	

Figure 12-13: Illustrating the grouping of MPM incidents for the two patient records

REGIMEN And DRUGS FACT Record 17		*for Patient Key ZU00031513*
Regimen Date:	20070524	
Captured On:	DRFOLUP	
ARV Regimen:	1b	
Drugs (adult):	3TC D4T NVP	
REGIMEN And DRUGS FACT Record 18		*for Patient Key ZU00029180*
Regimen Date:	20070604	
Captured On:	NURSEFOLUP	
ARV Regimen:	1b	
Drugs (adult):	3TC D4T NVP	

Figure 12-14: Illustrating the grouping of Regimen for the two patient records

Figure 12-15: Illustrating the Home Affairs linkage result

12.9. Evaluating the Longitudinal Patient Record

All LPR requests were logged to a logging table called LPR_USAGE (see figure 12-16). The purpose of this table was to provide an auditing trail of LPR requests but at the same time provide data for analyzing LPR request patterns.

COLUMN ID	Column Name	Data Type
1	LPR_USAGE_ID	NUMBER
2	MTUSERNAME	VARCHAR2(50 BYTE)
3	USERNAME	VARCHAR2(100 BYTE)
4	PERSAL	VARCHAR2(100 BYTE)
5	USAGE_PATIENT_KEY	VARCHAR2(20 BYTE)
6	USAGE_IP	VARCHAR2(20 BYTE)
7	USAGE_DATETIME	DATE

Figure 12-16: LPR_USAGE table

The final two phases of action research are to *evaluate* and undertake *specifying learning*. At the time of writing this chapter, insufficient data (< 25 rows) was available in the audit table to allow for a comprehensive pattern analysis. Top Management of the FSDOH was also in the process of approving the LPR, and it was therefore not implemented at each ARV treatment hospital or clinic. However, in order to test its basic functionality, the LPR was made available to the FSDOH research partners situated at the UCT Lung Institute and the MRC. At the same time the developers of the Meditech MPM application, LSSData situated in Boston USA, was also provided with a prototype. These researchers were instrumental in the data warehouse evaluation phase and suggested the integration of data to provide a view over time of a patient's clinical encounters.

A Senior Manager of Biomedical Informatics Research at the MRC had the following to say: "Thank you very much for developing the Longitudinal Patient Record (LPR) application and making it available to us. This is a ***major achievement*** and a substantial help to our efforts to strengthen service delivery for patients accessing the Free State public sector antiretroviral treatment program"

He also pointed out several key features which will be used as part of a research program to implement treatment failure clinics in the Free State. These key features can be summarized as follows:

- The web accessible front end is very useful and will enable researchers to access data from any site with internet access.
- The structuring of the data in terms of events, i.e. an encounter-based system is extremely important and will enable researchers to use it for cohort analysis.
- The integration of longitudinal patient data from patients registered on the ART program with other data sets, including TB and Home Affairs is extremely useful.

He concluded by stating that this data resource is unsurpassed in terms of its ability to support cohort analysis and that the MRC hopes to also develop a patient failure summary sheet on top of this data. He also pointed out that he believes the integration of ART, TB and the Home Affairs vital registration system to be a ***first in South Africa***.

A Senior Researcher of the Free State Antiretroviral Therapy programme at the UCT Lung Institute had the following to say: “I have just spent a wonderful hour working on the LPR. It is absolutely superb! I think you have really done something incredible innovative. A ***first for South Africa*** or ***Africa*** for that matter” Four minor suggestions were offered on how to improve the functionality of the LPR and also patient confidentiality. All of the suggestions were addressed.

The chief developer of the Meditech MPM application in the United States had the following to say: “Thanks for sharing this info regarding the LPR project and the data warehouse! From the PDF you attached it's easy to see the value of this project! I'm sure this took a lot of work to put together; congratulations to this achievement!”

12.10. Chapter Summary

This chapter examined and documented the use of the mapping table (containing all the probabilistic linkage results) to construct a coherent longitudinal electronic patient record. The chapter defined the definition of a longitudinal patient record (LPR), which in turn was used as basis to construct the LPR algorithm. The LPR algorithm was tested and deployed on a web based application as interface to the user. Additional security measures were developed to ensure only authorized users can query a patient’s LPR.

Finally an auditing table was attached to the LPR which in turn can be used to monitor and evaluate the usage of the LPR application. Even though insufficient data was available for a scientific analysis and evaluation, subjective impressions were obtained from the FSDOH research partners and the developers of the Meditech MPM application. These impressions could be viewed as a good indicator that the LPR was moving in the right direction and that it would be very useful in ART cohort analysis.

CHAPTER 13 – SUMMARY AND CONCLUSIONS

13.1. Introduction

The AIDS epidemic, caused by the Human Immunodeficiency Virus (HIV) is a global crisis which threatens development gains, economies, and societies. Within sub-Saharan Africa, where the epidemic began the earliest and the HIV prevalence is the highest, African countries have death rates not seen since the 1950's or 1960's. Sub-Saharan Africa remains the most heavily affected by HIV, accounting for 67% of all people living with HIV and for 72% of AIDS deaths in 2007. Worldwide, almost 20 million people are dead and 33 million people are living with HIV, with Sub-Saharan Africa totalling 14.4 million (72%) deaths and 22.11 million (67%) living with HIV.

The eradication of HIV/AIDS represents one of humanity's greatest challenges, which requires co-operation, and comprehensive collaboration between science, governments, social institutions, the media, the professions, and the general public. In this endeavour strategic information plays a major role. The following sections will summarize how the necessary strategic information was obtained to assist the FSDOH in dealing with the HIV and AIDS epidemic.

13.2. Aim and Objective

The Free State Department of Health introduced Antiretroviral Therapy (ART) to meet the needs of the rising number of people living with HIV and also dying due to AIDS related diseases. A patient information system was deployed by the Province to supplement the ART programme rollout process by gathering data and providing all the basic patient antiretroviral information. The lack of strategic information was prominent if one takes a closer look at the Free State antiretroviral treatment programme rollout and supporting patient information system. The patient information system was a traditional online clinical system, dealing with the bread-and-butter issues of accumulating data on a patient. Very little functionality was provided to deal with the complexities of managing the clinical outcomes of the ART programme. To add to the problem, other operational systems had to be interrogated to gain an understanding of the impact the rollout of ARV had. These operational systems ranged from standalone Human Resource systems to information systems accumulating data on tuberculosis which is closely related to HIV/AIDS. No mechanism or platform existed to provide management with integrated strategic information to manage the business process intelligently.

This study focussed on the challenges and solutions to overcome this shortfall. The main objective of this study was to supply comprehensive integrated strategic information for the management of the ART programme in the Free State Department of Health (FSDOH). The main objective was reached by means of a phased approach.

13.3. Research Design

The research methodology followed by this study was action research. For this study the researcher and the practitioner was the author of this thesis. The study repeated the action research methodology over two phases. Phase 1 covered the construction of a data warehouse using independent data marts in order to provide strategic information to management. Phase 2 covered the shortcomings of phase one and addressed those by integrating the independent data marts and providing a longitudinal patient record as end result that delivered comprehensive integrated strategic information.

13.3.1. Action Research - Phase One

Phase 1 of the action research commenced in Chapter 2 with a *problem diagnosis* phase. The worldwide AIDS epidemic was discussed, then the Sub-Saharan picture and finally the picture in the Free State. The ART programme that was introduced in the Free State to assist in combating the devastating effects of the AIDS disease was dealt with in detail. Chapter 2 concluded with the Free State model of care, which is at the heart of the Clinic Information System that would be implemented.

Chapter 3 expanded on the *problem diagnosis phase* which started in Chapter 2 and conceptualized the role of the paper-based forms, Clinic Information System and the two separate data warehousing projects. The main *problem identified* was to develop a single data warehouse for the FSDOH. One of the challenges was to incorporate the standalone ARV data warehouse (developed by the MRC) as a data mart into the new data warehouse. Other data marts such as human resources management and revenue collection were also required and had to be developed. The end product was a single "appropriately designed" data warehouse made available for operation to the relevant Departments in the FSDOH. A decision was made to develop the data warehouse in two phases. A theoretical framework was proposed which clearly described the two different action research cycles that was followed to *actively* solve the problem. A roadmap on how Phase 1 would concentrate on constructing a low-cost data warehouse with independent data marts and business intelligence solution to gain business buy-in was provided. Similarly, a roadmap on how Phase 2 would concentrate on integrating all the data-marts into a single longitudinal patient record was provided.

Chapter 4 provided a theoretical overview of business intelligence and data warehousing. A theoretical data warehouse architecture was proposed as a combination of the Inmon and Kimball methodology. This approach was adopted due to the lack of the necessary resources required to build a top-down enterprise healthcare model and a central data warehouse. Chapter 4 formed part of the *action planning* of Phase 1.

Chapter 5 introduced key database features of Oracle 10g that would be used to maximize the flexibility and performance of the FSDOH data warehouse. These features included data partitioning, query optimization, summarization, materialized views, bitmap indexes and parallelism (DML and queries). All of these features were used during the construction of the FSDOH data warehouse. Chapter 5 also formed part of the *action planning* of cycle one.

Chapter 6 of cycle one covered the *action taking* phase. Several data marts were developed in the FSDOH data warehouse. The first data mart was the Human Resource Data Mart (**HRDM**) whose primary goal was to provide the human resource development and management business unit with the necessary strategic information to perform workforce management for the entire FSDOH. The second data mart was the Antiretroviral Human Resources Data Mart (**ARVHRDM**) whose primary goal was to provide the managers in the ART programme with strategic workforce management information on staff that was involved in the ART programme only. In essence this data mart was a subset of the HRDM, but focused on the staff and facilities involved in ART and not the entire FSDOH. The third data mart was the Patient Admissions and Debiting Data Mart (**PADSDM**) whose primary goal was to gather strategic information relevant to revenue collection. With the intervention of integrating the ARV data warehouse, a secondary goal was identified to gather and link all clinical information of ARV patients that were admitted into hospitals for non-ARV treatment. The forth data mart was the conversion of the standalone ARV Data Warehouse (developed by the MRC) into a data mart. This data mart was then integrated into the FSDOH data warehouse. This data mart was called the Antiretroviral Therapy Data Mart (**ARVDM**). Its primary goal was to provide internal management reports as well as external reports for acquiring funding. The fifth data mart was the Tuberculosis Data Mart (**TBDM**) whose primary goal was to provide strategic information of the TB programme of the province. The final data mart was the Notifiable Diseases Data Mart (**NDDM**) whose primary goal was to provide strategic information from the notifiable diseases transactional system in terms of tuberculosis and diarrhea which are both closely associated with HIV and AIDS. Two of the main three dimensions were also conformed, namely Locations and Dates. The Patient dimension was the only dimension table that was non-conformed amongst all the data marts. This was due to the fact that all the data sources used different patient tables and it would have required a major effort to create a standardized patient table at that stage. In order to get an operational data warehouse up and running as soon as possible, this task was postponed for Phase 2.

Chapter 7 expanded on the work done in Chapter 6 and covered the construction of a Cognos 8 business intelligence solution. Multiple cubes and numerous ad-hoc reports were developed for each of the data marts described in Chapter 6. Chapter 7 also formed part of the *action taking* phase of Phase 1.

Chapter 8 covered the *evaluation* phase of Phase 1. A questionnaire was developed and distributed amongst data warehouse users. All users received a questionnaire if they were either using the data warehouse themselves or request information from the knowledge workers who extract information

for them from the data warehouse on a regular basis. The findings were extremely positive, with 93.8% of the respondents indicating that the FSDOH data warehouse is a valuable asset. Two additional but valuable actions were identified in the evaluation phase. These actions were documented as part of the *specifying learning* phase. The first action was to enrich the ARVDM with information on blood tests, linkage to the Vital Patient Register at Home Affairs and other data marts in the FSDOH (i.e. TBDM and NDDM). The second action was to combine all the information related to the ART programme into a single, integrated longitudinal patient record. Chapter 8 concluded with a conceptual framework of the longitudinal patient record, which formed the theoretical foundation and architecture of Phase 2.

13.3.2. Action Research - Phase Two

Chapter 8 provided the necessary input into for the *problem diagnosis* phase of Phase 2. Clearly, an *intervention* was required to add additional data marts and link up all these individual data marts and provide an **integrated longitudinal patient record.** Chapter 9 covered the *problem diagnosis* phase and discussed the creation of the additional data marts to enrich the ARVDM. Chapter 9 also provided a roadmap on how the additional data marts should be integrated in the FSDOH data warehouse bus architecture. The first additional data mart that was developed was the NHLS Blood Results Data Mart (**NHLSDM**). This data mart extracted data from an external data source associated with the National Health Laboratory Service (NHLS) and made it available in the FSDOH data warehouse. The second additional data mart that was developed was the Hospitalization Data Mart (**HOSPDM**). This data mart contained clinical inpatient information on all patients admitted to any of the four hospitals using the larger Meditech HIS application. The hospitalization information was extracted from the Meditech Admissions Module and the Meditech Outpatients Module (external data sources) and a data mart was created based on clinical encounters. The final additional data mart was not a data mart *per se.* The Home Affairs data mart included the mechanisms to extract and compare information from the National Population Register situated at the Department of Home Affairs. The findings of whether a patient was alive or deceased were then made available within the ARVDM.

Chapter 10 outlined the theory associated with record linkage and provided the basis of linking the data marts together in order to make a longitudinal record possible. A proposed record linkage strategy was introduced with a literature investigation on how this project was different from work already done in healthcare and antiretroviral therapy management. This chapter covered the *action planning* phase of Phase 2.

The *action taken* phase was covered in Chapter 11 and partially in Chapter 12. Chapter 11 outlined the plan to use probabilistic methods in linking up the relevant data fields from different data marts and external data sources to provide a coherent longitudinal electronic patient record for all ARV patients (waiting and on treatment). The following individual data marts were included in this plan:

PADSDM, ARVDM, TBDM, NDDM, NHLSDM and HOSPDM. Chapter 11 discussed in detail how both deterministic and probabilistic linkage methods were employed on each data mart individually, and then between all the data marts and the ARV data mart. The chapter concluded with the finding that the probabilistic linkage mechanisms performed better than the deterministic linkage mechanisms. By using this finding, only probabilistic linkage results were merged into a mapping table. This mapping table played a pivotal role in constructing the LPR.

Chapter 12 continued with the *action taken* phase of Phase 2. The mapping table with the probabilistic matching results of all the patient dimensions were implemented and documented in this chapter. A novel LPR algorithm was developed to combine all the relevant information together which provided an **integrated longitudinal patient record.** In order to make the LPR accessible for users, a web-based interface was developed and was accessible from in the Meditech MPM application. Chapter 12 concluded with the *evaluation* and *specifying learning* phases of Phase 2. Although no scientific evaluation was performed, due to the unavailability of sufficient data for analysis, impressions were provided by the FSDOH external research partners. Both these partners (MRC and UCT Lung Institute) were instrumental during the evaluation phase of Phase 1 and strongly suggested the construction of an integrated patient view. Their impressions were very positive about the usability and functionality of the LPR. It was pointed out that several key features of the LPR will be used as part of a research program to implement treatment failure clinics in the Free State. **See Appendix G.**

13.4. Research Conclusions

The main objective of this study was to construct a framework for providing strategic information for the management of the ART programme in the Free State Department of Health. This objective was reached via a two phase approach. During phase one a central data warehouse was designed and developed that incorporated an existing ARV data warehouse and several other independent operational sources, all related to ARV. This warehouse was evaluated by the users who overwhelmingly rated the warehouse as successful. From one system it was possible for managers to obtain strategic information on ARV encounters, ARV human resources, revenue collection, in-patient hospitalization, notifiable diseases and tuberculosis. This was achieved with a very limited budget and using in-house resource. Valuable experience was gained by the developer, which in turn, enabled the FSDOH to own the program management and skills development processes. These processes of BI competency were strongly emphasized at the 2008 Gartner Symposium. The success of the system opened the eyes of management to the contribution data warehousing can make to an organization and consequently obtained their buy-in for further development.

Although the data warehouse was a valuable asset by itself, several improvements were identified during the evaluation of the warehouse. The most important of these were to add additional data marts, to conform all the dimensions in order to obtain strategic information across the data marts and

lastly to be able to track an ARV patient over time over all facilities of care. This led to the second phase of the research. In this phase several data marts (NHLSDM, HOSPDM and linkage to Home Affairs) were firstly added. Secondly the patient dimensions of all the data marts were conformed through the process of probabilistic record linkage. Lastly a longitudinal patient record was developed that displayed all the encounters of an ARV patient over time. Even though the LPR could not be scientifically evaluated, the few that did test it rated it very highly and reckoned it was a first for South Africa.

This study, therefore, concludes with the fact that it was possible to construct a framework for providing comprehensive strategic information for the management of the ART programme in the Free State Department of Health. The research hypothesis as stated in Chapter one can, therefore, be accepted.

13.5. Contribution to the body of scientific knowledge

Research has been done previously with regards to the usage of data warehousing in the general health care field. Very little work has, however, been done on the usage of data warehousing to provide strategic information for the management of antiretroviral therapy, a problem mostly occurring in developing or poor countries. This study intended to fill this gap. The most important research contributions this study made in this endeavor are highlighted below:

- A theoretical framework was proposed to incorporate several independent operational systems, all related to ARV, into one central repository in order to provide strategic information on the ARV rollout.

- A mixture if Inmon and Kimball methodologies were used to construct a data warehouse based on independent data marts to provide strategic ARV information. In this whole process an existing ARV data warehouse was incorporated as a data mart.

- A BI solution was developed that made it easy for managers to interact with the data warehouse. Numerous cubes and reports were developed and made available to mangers in order to help in this regard. A define need for training of the managers was identified so that ad hoc queries can be performed on the data warehouse.

- A data warehouse evaluation was done by developing a questionnaire to test the success of the warehouse. Even though some of the questions came from previous studies, several new questions were added. Contrary to other studies, this study found that the users tended to use the data warehouse for structured tasks instead of unstructured tasks.

- To conform the patient dimensions of the different data marts, record linkage was used. Nowhere in the relevant literature could it be found, as far as possible, that this methodology was used for this purpose before. This also led to a new proposed ETL process of ETL plus record linkage.

- Probabilistic and deterministic methods were explored and tested on South African names and surnames. Contrarily to what was anticipated, no probabilistic matching problems were experienced when using African surnames and first names. Both Soundex and String comparison algorithms were used in the probabilistic matching and the results were good and useful.

- During the same file linkage process, numerous duplicate patients were identified. Methods were developed to de-duplicate the different dimensions. Firstly, all duplicates were identified with an internal linkage process. The outcome of this process provided the ability to group duplicate patient records together. This process was followed with the two-file record linkage process. The two-file probabilistic outcomes enabled the linkage of all de-duplicated dimension tables to the ARVDM by means of a mapping table. The integrated mapping table provided the necessary flexibility to link all the numerous facts (from the fact tables) with the de-duplicate dimension tables into a single dimensional model.

- A novel and real-world tested algorithm was developed to construct a longitudinal patient record from the integrated mapping table. The algorithm in turn provided the foundation for the LPR of an ART patient. A web based application was developed to provide a user interface to the LPR which was also linked and referenced by the OTLP system (Meditech MPM). This provided the users with a single point of entry, reducing confusion and instead increasing the availability of information on a patient. The ARV LPR is a first for South Africa and for that matter, the whole Africa.

- The work and research findings on the data warehouse and more specifically the HRDM, has already been used for the implementation of a country-wide human resource information system tender which was recently awarded to Oracle, South Africa. The author of this thesis is playing a critical role in developing the ETL interface for uploading data out of PERSAL into the National HRMIS database.

13.6. Further Research

Future research could be done to determine whether balanced score carding could assist with the management of antiretroviral therapy. The FSDOH has embarked on a business process engineering exercise to provide top management with information on key performance indicators (KPI). The functionally of BI dashboards and BI scorecards can meet this requirement.

Future research could be done to determine whether the longitudinal patient record could be updated more regularly and mechanisms can be explored to perform real-time probabilistic record linkage. This study provided a new proposed ETL process of “ETL plus record linkage”. A future challenge would be to make to this “ETL plus record linkage” process, real-time.

As was indicated previously, a research program to implement treatment failure clinics in the Free State was proposed by the external FSDOH research partners. Additional work was identified to integrate *virology study results* and *changes in ART regimens* into the LPR. These additional data sources will increase the accuracy of cohort analysis.

Patient confidentially in the LPR web application can be improved. As was previously indicated, patient confidentiality is an important issue when dealing with sensitive topics such as HIV and AIDS since it’s a non-disclosure disease. Possible solutions could be proposed which includes the usage of biometrics for authorization and authentication.

South Africa as a country is also embarking on the development of a nation-wide electronic health record (eHR.za project). The eHR.za project comprises of linking all nine provincial hospital information systems into a single country-wide electronic health record (EHR) architecture. A data warehouse and a range of smart card technologies are proposed to accomplish this challenge. This in turn would mean that smart cards would be freely available in the Free State province. The functionality of storing the LPR information (for the ART programme) onto an electronic smart card can be tested. These smart cards could then enable patients to carry their integrated ART information with them, as they move around South Africa, visiting public as well as private health care providers.

BIBLIOGRAPHY AND REFERENCES

NOTE: References designated with * have been referenced in the thesis. The other sources have been consulted, but not referenced.

1. Adam, F., Fahy, M. and Murphy, C. (1998). A framework for the classification of DSS usage across organizations. *Decision Support Systems*, 22(1), pp1 -13.

2. *AIDS Foundation of South Africa. (2006). Current Trends. Retrieved 2 February 2006 from the World Wide Web: http://www.aids.org.za/hiv.htm.

3. *Aldridge, D. (2004). A Practical Guide to Data Warehousing in Oracle. Retrieved 17 July 2006 from the World Wide Web: http://www.databasejournal.com/features/oracle/article.php/3353191.

4. Ananthakrishna, R., Chaudhuri, S. and Ganti, V. (2002). Eliminating Fuzzy Duplicates in Data Warehouses. *Proceedings of the 28th International Conference on Very Large Databases (VLDB 2002)*, pp586-597. Retrieved 13 February 2007 from the World Wide Web: http://www.vldb.org/conf/2002/S17P01.pdf.

5. *ASSA2003. (2005). The AIDS and Demographic Model. Retrieved 30 October 2008 from the World Wide Web: http://www.actuarialsociety.org.za/Portals/1/Documents/9f459fa2-4eeb-41ff-bee6-83b6e5e29b44.zip.

6. Bacher, S. (2007). The quest for zero defect. Computing SA February 2007, p17.

7. *Baer, H. (2005). Partitioning in Oracle Database 10g Release 2: An Oracle White Paper, May 2005. Retrieved 3 September 2006 from the World Wide Web: http://www.oracle.com/technology/products/bi/db/10g/pdf/twp_general_partitioning_10gr2_0505.pdf.

8. *Barbusinski, L., Howard, S., Jennings, M., Kelley, C. and Oates, J. (2003). The relationship between a fact and dimension table, DMReview.com. Retrieved 14 December 2006 from the World Wide Web: http://www.dmreview.com/article_sub.cfm?articleId=6349.

9. *Barrette, J. (2004). Achieving Human Capital Management: Building the Workforce Analytics Infrastructure. *DM REVIEW*, 14(4), pp22-25. Retrieved 7 July 2006 from the TOC Premier database.

10. *Baskerville, R.L. (1999). Investigating Information Systems with Action Research. *Communications of the Association for Information Systems*, 2(19). Retrieved 17 September 2007 from the World Wide Web: http://cais.isworld.org/articles/2-19/default.asp?View=pdf&x=78&y=17.

11. Baskerville, R.L. and Wood-Harper, A.T. (1998). Diversity in information systems action research methods. *European Journal of Information Systems*, 7(2), pp90-107.

12. *Becker, B. (2004). Kimball Design Tip #53: Dimensions Embellishments. Retrieved 3 September 2006 from the World Wide Web: http://www.rkimball.com/html/designtipsPDF/KimballDT53Dimension.pdf.

13. *Berndt, D.J. (2001). Consumer Decision Support Systems: A Health Care Case Study. *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, Retrieved 17 July 2006 from the TOC Premier database.

14. *Berndt, D.J. and Fisher, J.W. (2001). Understanding Dimension Volatility in Data Warehouses (or Bin There Done That). *Sixth INFORMS Conference on Information Systems and Technology* (INFORMS/CIST-2001). Retrieved 28 August 2006 from the World Wide Web: http://www.coba.usf.edu/berndt/research/papers/cist2001volatility_talk.pdf.

15. *Berndt, D.J., Hevner, A.R. and Studnicki, J. (2000). Community Health Assessments: A Data Warehousing Approach. *Proceedings of the 8th European Conference on Information Systems (ECIS 2000).*

16. *Berndt, D.J., Hevner, A.R. and Studnicki, J. (2003). The CATCH data warehouse: Support for community health care decision-making. *Decision Support Systems* 35(3), pp367-384.

17. *Bernillon, P., Lievre, L., Pillonel, J., Laporte, A. and Costagliola, D. (2000). Record-linkage between two anonymous databases for a capture-recapture estimation of underreporting of AIDS cases: France 1990-1993. The Clinical Epidemiology Group from Centres d'Information et de Soins de l'Immunodéficience Humaine. *International Journal of Epidemiology*, *29*(1), p168-174. Retrieved 2 February 2007 from the MEDLINE database.

18. *Beyer, M.A. (2005). Of Data Warehouses, Operational Data Stores, Data Marts and Data 'Outhouses'. Gartner Document ID Number: G00133092. Retrieved 29 October 2007 from Gartner Research database.

19. Blakely, T. and Salmond, C. (2002). Probabilistic record linkage and a method to calculate the positive predictive value. *International Journal of Epidemiology*, 31, pp1246-1252. Retrieved 8 December 2006 from the World Wide Web: http://ije.oxfordjournals.org/cgi/reprint/31/6/1246.

20. Blum, F. (1955). Action Research – A Scientific Approach? *Philosophy of Science*, 22(1), pp1-7.

21. *Bruckner, R.M. and Tjoa, A.M. (2002). Capturing Delays and Valid Times in Data Warehouses – Towards Timely Consistent Analyses. *Journal of Intelligent Information* Systems, 19(2), p169-190. Kluwer Academic Publishers.

22. *Butler, T., Feller, J., Pope, A., Murphy, C. and Emerson, B. (2006). An action research study on the design and development of core IT artifacts for knowledge management systems. Retrieved October 08, 2007 from the World Wide Web: http://csrc.lse.ac.uk/asp/aspecis/20060044.pdf.

23. *Canadian Centre for Occupational Health and Safety (CCOHS). (1997). AIDS in the Workplace. Retrieved 8 October 2007 from the World Wide Web: http://www.ccohs.ca/oshanswers/diseases/aids/aids.html.

24. *Carter, J.H. (2001). Electronic Medical Records: A Guide for Clinicians and Administrators. ACP Press. Retrieved 29 October 2008 from the World Wide Web: http://books.google.co.za/books?id=zBhXevV9GuEC.

25. Chaudhuri, S and Dayal. U. (1997). An overview of Data Warehousing and OLAP technology. *ACM SIGMOD Record*, 26(1), pp65-74. Retrieved 20 February 2007 from the World Wide Web: http://doi.acm.org/10.1145/248603.248616.

26. *Chapman, R.D. (2003). Plan for Implementation of ARV's in the Free State Province. Unpublished.

27. Chen, P.M., Lee, E.K., Gibson, G.A., Katz, R.H., Patterson, D.A. (1994). RAID: high-performance, reliable secondary storage, *ACM Computing Surveys (CSUR)*, 26(2), pp145-185. Retrieved 14 July 2006 from the World Wide Web: http://doi.acm.org/10.1145/176979.176981.

28. *Chen, R.Y., Accortt, N.A., Westfall, A.O., Mugavero, M.J., Raper, J.L., Cloud, G.A., Stone, B.K., Carter, J., Call, S., Pisu, M., Allison, J. and Saag, M.S. (2006). Health Care Costs for HIV Patients. *Clinical Infectious Diseases,* 2006 (42). Retrieved 29 June 2006 from the World Wide Web: http://www.natap.org/2006/HIV/022806_05.htm.

29. *Clark, D.E. (2004). Practical introduction to record linkage for injury research. *Injury Prevention Journal,* 2004 (10), pp186-191. Retrieved 20 February 2007 from the World Wide Web: http://ip.bmj.com/cgi/reprint/10/3/186.

30. *Cognos. (2006). Architecture and Planning Guide. Cognos Corporation, Wayside Road, Burlington, MA. Retrieved 18 January 2007 from the World Wide Web: http://support.cognos.com/

31. *Cohen, W.W., Ravikumar, P. and Fienberg, S.E. (2003a). A Comparison of String Distance Metrics for Name-Matching Tasks. *American Association for Artificial Intelligence*. Retrieved 23 January 2007 from the World Wide Web: http://secondstring.sourceforge.net/doc/iiweb03.pdf.

32. *Cohen, W.W., Ravikumar, P. and Fienberg, S.E. (2003b). A Comparison of String Metrics for Matching Names and Records. *American Association for Artificial Intelligence*. Retrieved 23 January 2007 from the World Wide Web: http://www.cs.cmu.edu/~wcohen/postscript/kdd-2003-match-ws.pdf.

33. *Cole, N. (2003). Feasibility and Accuracy of Record Linkage to Estimate Multiple Program Participation: Record Linkage Issues and Results of the Survey of Food Assistance Information Systems, E-FAN-03-008-1, Volume 4, pp1-78. United States Department of Agriculture, Economic Research Service. Retrieved 21 August 2006 from the World Wide Web: http://www.ers.usda.gov/publications/efan03008/efan03008-1/efan03008-1b.pdf.

34. *Corr, L. (2001). Kimball Design Tip #17: Populating Hierarchy Helper Tables. Retrieved 22 September 2006 from the World Wide Web: http://www.kimballgroup.com/html/designtipsPDF/DesignTips2001/KimballDT17Populating.pdf.

35. *Cyran, M. and Lane, P. (2003). Oracle Database Concepts, 10g Release 1 (10.1). Part No. B10743-01, Oracle Corporation, Redwood City, CA, USA.

36. *Davis, X., Wan, C., Ross, L., Wen, X. and Thomas, B. (2002). A data warehouse concept for HIV prevention program evaluation. *AIDS Education and Prevention: Official Publication of the International Society for AIDS Education*, 14(3 Suppl A), pp120-122.

37. *De Beer, E. (2006). Mobilising BI with data quality. Computing SA July 2006, p35.

38. *DeGruy, K.B. (2000). Healthcare Applications of Knowledge Discovery in Databases. *Journal of Healthcare Information Management,* 14(2), pp59-69. Retrieved 24 July 2006 from the World Wide Web: http://www.4reach.com/articles/hbi/HApplicationsKDD.pdf.

39. Dey, D., Sarkar, S., and De, P. (1998). A Probabilistic Decision Model for Entity Matching in Heterogeneous Databases. *Management Science*, 44(10), pp1379-1395. Retrieved 12 June 2007 from the Business Source Complete database.

40. Dick, R.S., Steen, E.B. and Detmer, D.E. (1991). The Computer-Based Patient Record: An Essential Technology for Health Care. Institute of Medicine. National Academy Press, Washington, DC, USA.

41. *Dodge, G. and Gorman, T. (2000). Essential Oracle8i Data Warehousing: Designing, Building and Managing Oracle Data Warehouses. Wiley. New York.

42. *Dorrington, R.E., Johnston, L., Bradshaw, D. and Danel, T.J. (2006). The Demographic Impact of HIV/AIDS in South Africa: National Indicators for 2006. Cape Town: Centre for Actuarial Research, South African Medical Research Council and Actuarial Society of South Africa. Retrieved 31 October 2008 from the World Wide Web: http://www.mrc.ac.za/bod/DemographicImpactHIVIndicators.pdf.

43. *Ebidia, A., Mulder, C., Tripp, B. and Morgan, M.W. (1999). Getting data out of the electronic patient record: critical steps in building a data warehouse for decision support. *Proceedings of the 1999 AMIA Annual Symposium,* pp745-749. Retrieved the 27 July 2006 from the World Wide Web: http://www.amia.org/pubs/symposia/D005548.PDF.

44. *Eichhorst, B. (2002). Patient-centric HIS. A healthcare information system based on a longitudinal patient record provides benefits to patients--and clinicians, administrators and IT staff as well. *Health Management Technology*, *23*(4), pp40-42. Retrieved 29 May 2007 from Business Source Complete database.

45. *Ewen, E.F., Medsker, C.E. and Dusterhoft, L.E. (1998). Data warehousing in an integrated health system: building the business case. *Proceedings of the 1st ACM International Workshop on Data Warehousing and OLAP*. DOLAP '98. ACM Press, New York, pp47-53. Retrieved 13 June 2006 from the World Wide Web: http://nas.cl.uh.edu/boetticher/ML_DataMining/p47-ewen.pdf.

46. *Fair, M.E. (1997). Record linkage in an information age society. *Proceedings of an International Workshop and Exposition*, pp 427-441. Retrieved 4 September 2008 from the World Wide Web: http://www.fcsm.gov/working-papers/marthafair.pdf.

47. *Fair, M.E. (2004). Generalized Record Linkage System – Statistics Canada's Record Linkage Software. *Austrian Journal of Statistics*, 33(1&2), pp37-53.

48. Fellegi, I.P. and Sunter, A.B. (1969). A Theory for Record Linkage. *Journal of the American Statistical Association*, 64, pp1183-1210.

49. *Fennessy, G. and Burstein, F. (2000). Using soft systems as a methodology for researching knowledge management problems. *Proceedings from the 1st International Conference on Systems Thinking in Management, 2000*, pp180-185. Retrieved 8 October 2007 from the World Wide Web: http://ftp.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-72/026%20Fennessy%20SSM.pdf.

50. *Free State Department of Health (FSDOH). (2006). Corporate Strategic Plan 2006/2007 to 2014/2015. Unpublished.

51. *Fuchs, G. (2005). Reality IT: OLAP Schmolap. DM Review Online. Retrieved the 27 July 2006 from the World Wide Web: http://www.dmreview.com/article_sub.cfm?articleId=1020865.

52. *Gatziu, S. and Vavouras, A. (1999). Data warehousing: Concepts and mechanisms. *INFORMATIK*, 1, pp8-11.

53. *Golfarelli, M., Rizzi, S. and Cella, I. (2004). Beyond data warehousing: What's next in business intelligence? *Proceedings of the 7th ACM international workshop on Data warehousing and OLAP*. New York, pp1-6. Retrieved 28 November 2006 from the World Wide Web: http://doi.acm.org/10.1145/1031763.1031765.

54. Giorgini, P., Rizzi, S. and Garzetti, M. (2005). Goal-Oriented Requirement Analysis for Data Warehouse Design. *International Workshop on Data Warehousing and OLAP 2005*. Retrieved 4 July 2007 from the World Wide Web: http://www.troposproject.org/papers_files/dolap05.pdf.

55. *Grannis, S., Overhage, J., Hui, S. and McDonald, C. (2003). Analysis of a Probabilistic Record Linkage Technique without Human Review. *AMIA Annual Symposium Proceedings / AMIA Symposium,* pp259-263. Retrieved 26 January 2007 from the World Wide Web: http://www.pubmedcentral.nih.gov/articlerender.fcgi?tool=pubmed&pubmedid=14728174.

56. *Grannis, S., Overhage, J. and McDonald, C. (2004). Real World Performance of Approximate String Comparators for use in Patient Matching. *Medinfo*, 11(Pt 1), pp43-47. Retrieved 6 March 2007 from the MEDLINE database.

57. *Graig, R. (1999). Is a Data Mart in Your future? Retrieved 1 October 2007 from the World Wide Web: http://findarticles.com/p/articles/mi_m0FOX/is_5_4/ai_54044730.

58. Gray, G.W. (2004). Challenges of Building Clinical Data Analysis Solutions. *Journal of Critical Care*, *December 2004,* 19(4), pp264-270.

59. *Green, M.A. and Bowie, M.J. (2004). Essentials of Health Information Management. Cengage Learning, 2004.

60. *Goldstein, J. and Larson, P. (2001). Optimizing Queries using Materialized Views: A Practical, Scalable Solution, ACM SIGMOD Record, v.30 n.2, pp331-342.

61. *Gorla, N. (2003). Features to consider in a data warehousing system. *Communications of the ACM,* 46(11), pp111-115.

62. *Grigioni, S., Saba, J., Dintruff, R., Pechevis, M., Delbos, V., Muyingo, S. and Ladner, J. (2004). Expanding access to antiretroviral therapy in sub-Saharan Africa: Lessons from 20 countries. Retrieved 28 August 2006 from the World Wide Web: http://www.axios-group.com/en/library/posters/AbujaAHCPPoster261105.pdf.

63. *Gu, L., Baxter, R.A., Vickers, D. and Rainsford, C. (2003). Record Linkage: Current Practice and Future Directions. Technical Report 03/83, April 2003, CSIRO Mathematical and Information Sciences, Canberra, Australia. Retrieved 13 February 2007 from the World Wide Web: http://research.cmis.csiro.au/rohanb/PAPERS/record_linkage.pdf.

64. Hernandez, M.A. and Stolfo, S.J. (1995). The Merge/Purge problem for Large Databases. *Proceedings of ACM SIGMOD 1995,* pp127-138. Retrieved 2 February 2007 from the World Wide Web: http://citeseer.ist.psu.edu/stolfo95mergepurge.html.

65. *Hobbs, L., Hillson, S., Lawande, S. and Smith, P. (2005). Oracle Database 10g Data Warehousing. First Edition. Elsevier Digital Press. Oxford, United Kingdom.

66. Howe, G.R. and Linday, J. (1981). Generalized Iterative Record Linkage Computer System for Use in Medical Follow-up Studies. *Computers and Biomedical Research*, 14, pp327-340.

67. *Howe, G.R. (1998). Use of Computeried Record Linkage in Cohort Studies. *Epidemiologic Reviews*, 20(1), pp112-121.

68. *Hristovski, D., Rogac, M., Markota, M. (2000). Using Data Warehousing and OLAP in Public Health Care. *American Medical Informatics Association*, 7, pp369-373.

69. Hwang, M.I. and Xu, H. (2007). The Effect of Implementation Factors on Data Warehousing Success: An Exploratory Study. *Journal of Information, Information Technology, and Organizations*, 2007, 2, pp1-14. Retrieved 29 October 2007 from the World Wide Web: http://jiito.org/articles/JIITOv2p001-014Hwang40.pdf.

70. *Inmon, W.H. (2005). Building the Data Warehouse. Forth Edition. Wiley. Indianapolis.

71. *Jaro, M.A. (1995). Probabilistic linkage of large public health data files. *Statistics in Medicine [Stat Med]*, 14(5-7), pp491-498.

72. *Jin, L., Li, C. and Mehrotra, S. (2002). Efficient Record Linkage in Large Data Sets. UCI Technical Report, February 2002. Retrieved 13 February 2007 from the World Wide Web: http://www.ics.uci.edu/~chenli/pub/strjoin.pdf.

73. *Kanabus, A. (2006). AIDS & HIV antiretroviral drug treatment in resource poor communities. Retrieved 2 March 2006 from the World Wide Web: http://www.avert.org/safricastats.htm.

74. *Kelly, J. (2008). Gartner to CIOs: Beware these 'fatal' business intelligence flaws. Gartner Symposiom, 2008. Retrieved the 28 October 2008 from the World Wide Web: http://searchdatamanagement.techtarget.com/news/article/0,289142,sid91_gci1335728,00.html?track=NL-340&ad=672956&asrc=EM_NLN_4933971&uid=1072349.

75. *Kimball, R., Reeves, L., Ross, M. and Thornthwaite, W. (1998). The Data Warehouse Lifecycle Toolkit. First Edition. Wiley Computer Publishing. New York.

76. *Kimball, R. (1998a). Surrogate Keys – Keep control over record identifiers by generating new keys for the data warehouse. *DBMS*, May 1998. Retrieved 5 October 2006 from the World Wide Web: http://www.dbmsmag.com/9805d05.html.

77. *Kimball, R. (1998b). Bringing up Supermarts. *DBMS*, January 1998. Retrieved 13 May 2007 from the World Wide Web: http://www.dbmsmag.com/9801d14.html.

78. *Kimball, R. (1998c). Help for Hierarchies - Helper tables handle dimensions with complex hierarchies. *DBMS*, September 1998. Retrieved 18 December 2006 from the World Wide Web: http://www.dbmsmag.com/9809d05.html.

79. *Kimball, R. (1999). Stirring Things Up. *Intelligence Enterprise Magazine*, June 1999, 2(9). Retrieved 31 May 2007 from the World Wide Web: http://www.intelligententerprise.com/db_area/archives/1999/992206/warehouse.jhtml?_requestid=523455.

80. *Kimball, R. and Ross, M. (2002). The Data Warehouse Toolkit. Second Edition. Wiley Computer Publishing. New York.

81. *Knutsen, L. (2005). Optimizing your Data Warehouse Design for Superior Performance. *Hyperion Global Conference,* 2005. Retrieved 10 October 2007 from the World Wide Web: http://www.advancedatatools.com/TechInfo/2100A_Lester_Knutsen_FINALv1.pdf.

82. *Lane, P. and Lumpkin, G. (2000). *Oracle8i* Data Warehouse Guide Release 2 (8.1.6). Part No. A76994-01, Oracle Corporation, Redwood City, CA, USA.

83. *List, B., Bruckner, R.M., Machaczek, K. and Schiefer, J. (2002). A Comparison of Data Warehouse Development Methodologies Case Study of the Process Warehouse. *Proceedings of the 13th International Conference on Database and Expert Systems Applications,* 2002. Retrieved 02 October 2007 from the World Wide Web: http://www.ifs.tuwien.ac.at/~js/download/dexa02.pdf.

84. Lane, P., Schupmann, V. and Stuart, I. (2003). Oracle Database Data Warehousing Guide, 10*g* Release 1 (10.1). Part No. B10736-01, Oracle Corporation, Redwood City, CA, USA.

85. *Lau, R.K.W. and Catchpole, M. (2001). Improving data collection and information retrieval for monitoring sexual health. *International Journal of STD and AIDS*, 12(1), pp8-13.

86. *Lawyer, J. and Chowdhury, S. (2004). Best Practices in Data Warehousing to Support Business Initiatives and Needs. *Proceedings of the 37th Hawaii International Conference on System Sciences*, 2004. Retrieved 28 November 2006 from the World Wide Web: http://csdl.computer.org/comp/proceedings/hicss/2004/2056/08/205680223a.pdf.

87. *Li, B., Quan, H., Fong, A. and Lu, M. (2006). Assessing record linkage between health care and Vital Statistics databases using deterministic methods. *BMC Health Services Research*, *6*, pp48-58. Retrieved 13 February 2007 from the World Wide Web: http://www.biomedcentral.com/1472-6963/6/48.

88. *Liu, S. and Wen, S.W. (2000). Development of Record Linkage of Hospital Discharge Data for the Study of Neonatal Readmission. *Chronic Diseases in Canada*, 20(2). Retrieved 1 June 2007 from the World Wide Web: http://www.phac-aspc.gc.ca/publicat/cdic-mcc/20-2/c_e.html.

89. *Mailvaganam, H. (2007). Design Methodologies of Kimball and Inmon Plus a Third Way. Retrieved 2 October 2007 from the World Wide Web: http://www.dwreview.com/Articles/KimballInmon.html.

90. *Malinowski, E. and Zimanyi, E. (2004). OLAP Hierarchies: A Conceptual Perspective. *16th Int. Conf. on Advanced Information Systems Engineering (CAiSE)*, pp477–491. Retrieved 3 July 2007 from the World Wide Web: http://code.ulb.ac.be/dbfiles/MalZim2004inproceedings.pdf.

91. *Malinowski, E. and Zimanyi, E. (2006). Hierarchies in a Multidimensional Model: From Conceptual Modeling to Logical Representation. *Data & Knowledge Engineering,* 59(2), pp348-377. Retrieved 1 July 2007 from the World Wide Web: http://code.ulb.ac.be/dbfiles/MalZim2006article.pdf.

92. *Mallach, E.G. (2000). Decision Support and Data Warehouse Systems. International Edition 2000. McGraw-Hill Book Co. Boston.

93. *March, T.S. and Hevner, A.R. (2007). Integrated Decision Support Systems: A data warehousing perspective. *Decision Support Systems*, 43(3). Retrieved 4 June 2007 from the World Wide Web: http://mis.temple.edu/sigdss/icis03/proceedings/DSSWorkshop03-March.pdf.

94. Memel, D.S., Scott, J.P., McMillan, D.R., Easton, S.M., Donelson, S.M., Campbell, G., Sheehan, M. and Ewing, T.N. (2001). Development and implementation of an information management and information technology strategy for improving healthcare services: a case study. *Journal of Healthcare Information Management: JHIM*, 15(3 (Print)), pp261-285. Retrieved 15 March 2006 from the World Wide Web: http://www.himss.org/content/files/him15307_9612.pdf.

95. *Millsap, C.V. (1996). Configuring Oracle Server for VLDB. Oracle System Performance Group Technical Paper, Oracle Corporation. Retrieved 15 September 2006 from the World Wide Web: http://www.miracleas.com/BAARF/0.Millsap1996.08.21-VLDB.pdf.

96. *Muse, A.G, Mikl, J. and Smith, P.F. (1995). Evaluating the quality of anonymous record linkage using deterministic procedures with the New York State AIDS registry and a hospital discharge file. *Statistics in Medicine [Stat Med],* 14(5-7), pp499-509. Retrieved 2 February 2007 from the MEDLINE database.

97. *Nakache, D. (2003). Problems in Designing Hugh Data Warehouses and Datamarts. *In Proceedings of the Ninth Americas Conference on Information Systems (AMCIS'03),* 4-6 August, 2003, Tampa, Florida, USA, pp2449-2459. Retrieved the 7 August 2006 from the World Wide Web: http://cedric.cnam.fr/PUBLIS/RC559.doc.

98. *Nanda, A. (2006). Partition Decisions. *Oracle Magazine*, October 2006. Retrieved 6 December 2006 from the World Wide Web: http://www.oracle.com/technology/oramag/oracle/06-sep/o56partition.html.

99. *National Department of Health, South Africa. (2003). Cabinet's decision on the operational plan for comprehensive care and treatment of people living with HIV and AIDS. Pretoria. South Africa. Retrieved 16 July 2008 from the World Wide Web: http://www.doh.gov.za/docs/pr/2003/pr1119.html.

100. National Department of Health, South Africa. (2004). South African Department of Health Study. Retrieved 15 March 2006 from the World Wide Web: http://www.avert.org/safricastats.htm.

101. *National Department of Health, South Africa. (2007). HIV Antenatal Prevalence Survey 2007. Retrieved 30 October 2008 from the World Wide Web: http://www.doh.gov.za/docs/reports/2007/antenatal/antenatal_report.pdf.

102. *Nemes, M.I.B., Carvalho, H.B. and Souza, M.F.M. (2004). Antiretroviral therapy adherence in Brazil. *AIDS 2004*, 18 (suppl 3), pp15-20.

103. Newcombe, H.B. (1967). Record Linking: The Design of Efficient Systems for Linking Records into Individual and Family Histories. *American Journal of Human Genetics*, 19(3), part I.

104. Newcombe, H.B., Kennedy, J.M., Axford, S.J. and James, A.P. (1959). Automatic Linkage of Vital Records. *Science*, 130, no 3381, pp954-959.

105. *NHIS/SA-Committee. (2005). The National Strategic Framework for eHR Implementation in South Africa. Department of Health, South Africa. Unpublished.

106. *NHIS/SA-Committee. (2006). National Health Care Management Information System of South Africa (NHC/MIS), Department of Health, South Africa. Retrieved 15 May 2006 from the World Wide Web: http://www.doh.gov.za/nhis/docs/nchmis.htm.

107. *Nitsch, D., Morton, S., DeStavola, B.L., Clark, H. and Leon, D.A. (2006). How good is probabilistic record linkage to reconstruct reproductive histories? Results from the Aberdeen Children of the 1950s study. *BMC Medical Research Methodology*, 6, pp15-24. Retrieved 23 February 2007 from the World Wide Web: http://www.biomedcentral.com/content/pdf/1471-2288-6-15.pdf.

108. *Orr, K. (1996). White Paper on Data Warehousing Technology. Retrieved 31 July 2007 from the World Wide Web: http://www.focuss.com.au/resources/Data%20Warehousing %20Technology.pdf.

109. Park, Y. (2006). An empirical investigation of the effects of data warehousing on decision performance. *Information and Management Journal*, 43(1), pp51-61.

110. *Ponniah, P. (2001). Data Warehousing Fundamentals: A Comprehensive Guide for IT Professionals, John Wiley & Sons, Inc.

111. *Rishel, W., Thomas, J., Handler, M.D. and Edwards, J. (2005). A Clear Definition of the Electronic Health Record. Gartner document G00130927. Retrieved 8 November 2008 from the World Wide Web: http://www.gartner.com/.

112. Rizzi, S. (2003). Open Problems in Data Warehousing: 8 Years Later*. In Proceedings of the 5th International Workshop on Design and Management of Data Warehouses (DMDW),* 2003. Retrieved the 3 July 2007 from the World Wide Web: http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-77/keynote.pdf.

113. *Roos, L., Wajda, A. and Nicol, J. (1986). The art and science of record linkage: methods that work with few identifiers. *Computers in Biology and Medicine*, 16(1), pp45-57. Retrieved 20 February 2007 from the MEDLINE database.

114. *Ross, M. (2003). Kimball Design Tip #48: De-clutter with Junk Dimensions. Retrieved 13 September 2006 from the World Wide Web: http://www.rkimball.com/html/designtipsPDF/ DesignTips2003/KimballDT48DeClutter.pdf.

115. *Saraceni, V., Da Cruz, M.M., Lauria, L.M. and Durovni, B. (2005). Trends and Characteristics of AIDS Mortality in the Rio de Janeiro City after the Introduction of Highly Active Antiretroviral Therapy. *The Brazilian Journal of Infectious Diseases 2005*, 9(3), pp209-215.

116. *Sauleau, E.A., Paumier, J.P. and Buemi, A. (2005). Medical record linkage in health information systems by approximate string matching and clustering. *BMC Medical Informatics and Decision Making*, 5, pp32-45. Retrieved 13 June 2007 from the World Wide Web: http://www.biomedcentral.com/1472-6947/5/32.

117. *Scheese, R. (1998). Data warehousing as a healthcare business solution. *Journal of the Healthcare Financial Management Association,* February 1998, 52(2), pp56-59. Retrieved 26 June 2006 from the MEDLINE database.

118. *Schubart, J. and Einbinder, J. (2000). Evaluation of a data warehouse in an academic health sciences centre. *International Journal of Medical Informatics, 60*, pp319-333.

119. Shams, K. and Farishta, M. (2001). Data Warehousing: Toward Knowledge Management. *Topics in Health Information Management*, 21(3), pp24-32.

120. *Shin, B. (2003). An Exploratory Investigation of System Success Factors in Data Warehousing. *Journal of the Association for Information Systems*, 4, pp141-170.

121. *Snyman, C.S., Boucher, P., Cloutier, S., Puvimanasinghe, J.A. and Ndwapi, N. (2007). Establishing a Data Warehouse for patients on ART in Botswana. Retrieved 31 May 2007 from the World Wide Web: http://www.sim.hcuge.ch/helina/50.htm.

122. *Srivastava, J. and Chen, P.Y. (1999). Warehouse creation -- a potential roadblock to data warehousing. *IEEE Transactions on Knowledge & Data Engineering*, 11(1), pp118-126. Retrieved 2 August 2006 from the World Wide Web: http://www.cse.mrt.ac.lk/ lecnotes/cs5901/readinglist/srivastava.pdf.

123. *Stackowiak, R., Rayman, J. and Greenwald, R. (2007). Oracle Data Warehousing and Business Intelligence Solutions. First Edition. Wiley Publishing Inc. Indianapolis.

124. *STATSSA (2007). Mid-year population estimates for 2007. Retrieved 30 October 2008 from the World Wide Web: http://www.statssa.gov.za/publications/.

125. *Statistics Canada. (2007). Generalized Record Linkage System (GRLS) – Concepts Guide, Version 4.5.03.

126. *Stewart, R. and Loveday, M. (2005). Public HAART Projects in South Africa – Progress to November 2004. Durban: Health Systems Trust. Retrieved 1 July 2007 from the World Wide Web: http://www.health-e.org.za/resources/haart_progress1104.pdf.

127. Stolba, N., Nguyen, T.M. and Tjoa, A.M. (2007). Towards a Data Warehouse Based Approach to Support Healthcare Knowledge Development and Sharing, IRMA 2007 Proceedings, IRMA Press, pp245-248.

128. Susman, G. and Evered, R. (1978). An Assessment of the Scientific Merits of Action Research. *Administrative Science Quarterly*, 23(4), pp582-603.

129. *Sweeney, R.J., Davis, R.J. and Jeffery, M. (2002). Case Study: Teradata Data Mart Consolidation Return no Investment at GST. Retrieved 1 October 2007 from the World Wide Web: http://www.kellogg.northwestern.edu/faculty/jeffery/htm/cases/Data%20Mart%20 Consolidation% 20ROI %20Case%20at%20GST.pdf.

130. Tejada, S., Knoblock, C.A. and Minton, S. (2001). Learning Object Identification Rules for Information Integration. *Information Systems Journal*, 26(8), p607-633. Retrieved 12 February 2007 from the World Wide Web: http://www.isi.edu/integration/papers/tejada01-is.pdf.

131. *Thornthwaite W. (2000). Kimball Design Tip #11: Accurate Counts within a Dimension. Retrieved 13 September 2006 from the World Wide Web: http://www.ralphkimball.com/html/ designtipsPDF/DesignTips2000%20/KimballDT11AccurateCounts.pdf.

132. *UNAIDS. (2004). Report on Global AIDS epidemic, July 2004. Retrieved 1 February 2006 from the World Wide Web: http://www.unaids.org/bangkok2004/report_pdf.html.

133. *UNAIDS. (2006). UNAIDS/WHO AIDS epidemic update. Geneva, Switzerland. Retrieved 16 July 2008 from the World Wide Web: http://www.unaids.org/en/HIV_data/epi2006/default.asp.

134. *UNAIDS. (2008a). Report on the global AIDS epidemic 2008 – Executive summary. Retrieved 30 October 2008 from the World Wide Web: http://www.unaids/org/en/ KnowledgeCenter/HIVData/GlobalReport/2008/.

135. *UNAIDS. (2008b). HIV and AIDS Estimates and data, 2007 and 2001. Annexure 1. Retrieved 30 October 2008 from the World Wide Web: http://www.unaids/org/en/ KnowledgeCenter/HIVData/GlobalReport/2008/.

136. *UNAIDS. (2008c). Status of the global HIV epidemic. Annexure 2. Retrieved 30 October 2008 from the World Wide Web: http://www.unaids/org/en/KnowledgeCenter/HIVData/ GlobalReport/2008/.

137. *University of Cape Town, Medical Research Council, Free State Department of Health. (2004(1)). Implementation of the Comprehensive Care, Management and Treatment of HIV and AIDS Patients. Unpublished.

138. *University of Cape Town, Medical Research Council, Free State Department of Health. (2004(2)). Free State Anti-Retroviral Treatment Programme: Collection and Analysis of Routine Data. Unpublished.

139. *University of Cape Town, University of the Free State, Medical Research Council, Free State Department of Health. (2006(1)). Implementation of the Comprehensive Care, Management and Treatment of HIV and AIDS Programme 2005 Fourth Quarter Report. Unpublished.

140. *Van Bemmel, J.H. and Musen, M.A. (1999). Handbook of Medical Informatics, Website v3.3. Retrieved 17 November 2008 from the World Wide Web: http://www.mieur.nl/mihandbook/r_3_3/handbook/home.htm

141. *Verma, R. and Harper, J. (2001). Life Cycle of a Data Warehousing Project in Healthcare. *Journal of Healthcare Information Management,* 15(2), pp107-117. Retrieved 15 March 2006 from the World Wide Web: http://www.himss.org/content/files/jhim/15-2/him15204.pdf.

142. Vesset, D. (2006). Worldwide Data Warehousing Tools 2005 Vendor Shares, IDC #203229, 1. Retrieved 1 December 2006 from the World Wide Web: http://www.oracle.com/ corporate/analyst/reports/infrastructure/bi_dw/idc-dw-tools-2005-1.pdf.

143. *Whitehorn, M. (2007). Business intelligence and data warehouse trends: Appliances and "third-generation" BI (Part 1). Retrieved 14 September 2007 from the World Wide Web: http://go.techtarget.com/r/2191117/1072349.

144. *Winkler, W. E. (1990). String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage, *Proceedings of the Section on Survey Research Methods*, *American Statistical Association*, pp354-359.

145. *Winkler, W.E. (1999). The State of Record Linkage and Current Research Problems. Technical Report, Statistical Research Division, U.S. Census Bureau, 1999. Retrieved 26 January 2007 from the World Wide Web: http://www.census.gov/srd/papers/pdf/rr99-04.pdf.

146. *Winkler, W.E. (2005). Overview of Record Linkage and Current Research Directions. Research Report, Statistical Research Division, U.S. Bureau of the Census, 2005. Retrieved 12 February 2007 from the World Wide Web: http://www.census.gov/srd/papers/pdf/ rrs2006-02.pdf.

147. *Wixom, B. and Watson, H. (2001). An emperical investigation of the factors affecting data warehousing success. *MIS Quaterly*, 28(1), pp17-41.

148. *Wood, M. (2000). Changing the Rules in Enterprise Data Warehouse. *Health Management Technology*, 21(9), pp56-58. Retrieved 12 June 2007 from the Business Source Complete database.

149. *World Health Organization. (2005). Summary Country Profile for HIV/AIDS Treatment Scale-Up in Haiti, June 2005. Retrieved 28 August 2006 from the World Wide Web: http://ww.who.int/3by5/support/june2005_hti.pdf.

150. *Wu, M. and Buchman, A.P. (1997). Research Issues in Data Warehousing. *Datenbanksysteme in Bro, Technik und Wissenschaft (BTW)*, pp61-82. Retrieved 27 July 2005 from the World Wide Web: http://www.informatik.tu-darmstadt.de/DVS1/staff/wu/btwllncs.ps.gz.

Appendix A

```
SELECT MAX(DISTINCT EXTRACT_DATE)
INTO highest_month
FROM COMPONENT_STRUCTURE;

DELETE /*+ NOLOGGING */ FROM TEMP_ORGANOGRAM;

   INSERT /*+ APPEND NOLOGGING */ INTO TEMP_ORGANOGRAM
   SELECT
     a.EXTRACT_DATE,
     a.STATUS,
     sys_connect_by_path(a.COMPONENT, '/' )        AS SCBP,
     LEVEL                                         AS COMPONENT_LEVEL,
     a.COMPONENT,
     a.COMPONENT_NAME,
     a.CONTROL_COMPONENT,
     a.CONTROL_COMPONENT_NAME,
     NULL,
     a.RESPONSIBILITY,
     a.RESPONSIBILITY_NAME,
     NULL,
     NULL
   FROM (SELECT
         CTRL_COMPONENT.EXTRACT_DATE,
         CTRL_COMPONENT.STATUS,
         COMPONENT_TYPE,
         COMPONENT,
         CONTROL_COMPONENT,
         CONTROL_COMPONENT_DECODED        AS CONTROL_COMPONENT_NAME,
         RESPONSIBILITY,
         COMPONENT_DECODED                AS COMPONENT_NAME,
         RESPONSIBILITY_DECODED           AS RESPONSIBILITY_NAME
         FROM COMPONENT_STRUCTURE,
                  (SELECT extract_date, status, tableno, code, DESC_ENG_LONG
                   FROM LOOKUPCODES
                   WHERE tableno='810') CTRL_COMPONENT
         WHERE CTRL_COMPONENT.CODE = COMPONENT
         AND   CTRL_COMPONENT.EXTRACT_DATE = COMPONENT_STRUCTURE.EXTRACT_DATE
         AND   COMPONENT_STRUCTURE.EXTRACT_DATE=i.EXTRACT_DATE) a
   START WITH TRIM(a.COMPONENT) = '000004'
   CONNECT BY a.CONTROL_COMPONENT = PRIOR a.COMPONENT
   ORDER SIBLINGS by a.COMPONENT_NAME;

   COMMIT;
```

Appendix B

SQL code for make changes to all child components, in case a parent component had a change.

```
FOR j IN (SELECT
                    a.COMPONENT,
                    a.COMPONENT_LEVEL,
                    a.CONTROL_COMPONENT_NAME,
                     a.SCBP
              FROM HIERARCHY_ORGANOGRAM a
              WHERE a.EFFECTIVE_DATE_FROM <= i.EXTRACT_DATE
              AND (a.EFFECTIVE_DATE_TO >= i.EXTRACT_DATE
                     OR a.EFFECTIVE_DATE_TO IS NULL)
              MINUS
              SELECT
                    a.COMPONENT,
                    a.COMPONENT_LEVEL,
                    a.CONTROL_COMPONENT_NAME,
                    a.SCBP
              FROM TEMP_ORGANOGRAM  a
             ORDER BY COMPONENT_LEVEL) LOOP

    UPDATE TEMP_ORGANOGRAM
    SET CONTROL_COMPONENT_CHANGED='Y'
    WHERE CONTROL_COMPONENT_CHANGED IS NULL
    AND      COMPONENT IN (SELECT COMPONENT FROM temp_organogram a
                                START WITH TRIM(a.COMPONENT) = j.COMPONENT
                                CONNECT BY a.CONTROL_COMPONENT = PRIOR a.COMPONENT);
    COMMIT;

 END LOOP;
```

With the necessary changes made to the child components entries in TEMP_ORGANOGRAM, the two tables are compared for a final time with each other to detect all SCD Type 2 changes.

```
FOR j IN (SELECT
                 COMPONENT_LEVEL
               , COMPONENT
               , CONTROL_COMPONENT
               , SCBP
               , COMPONENT_NAME
               , CONTROL_COMPONENT_CHANGED
             FROM HIERARCHY_ORGANOGRAM a
             WHERE a.EFFECTIVE_DATE_FROM <= i.EXTRACT_DATE
             AND (a.EFFECTIVE_DATE_TO >= i.EXTRACT_DATE
                    OR a.EFFECTIVE_DATE_TO IS NULL)
             MINUS
             SELECT
                  COMPONENT_LEVEL
               , COMPONENT
               , CONTROL_COMPONENT
               , SCBP
               , COMPONENT_NAME
               , CONTROL_COMPONENT_CHANGED
             FROM TEMP_ORGANOGRAM a) LOOP
```

```
    UPDATE /*+ NOLOGGING */   HIERARCHY_ORGANOGRAM
    SET EFFECTIVE_DATE_TO=i.EXTRACT_DATE-1
    WHERE COMPONENT=j.COMPONENT
     AND EFFECTIVE_DATE_TO IS NULL;



    INSERT /*+ APPEND NOLOGGING */ INTO  HIERARCHY_ORGANOGRAM
    SELECT
            a.EXTRACT_DATE
          , a.STATUS
          , a.SCBP
          , a.COMPONENT_LEVEL
          , a.COMPONENT
          , a.COMPONENT_NAME
          , a.CONTROL_COMPONENT
          , a.CONTROL_COMPONENT_NAME
          , NULL
          , a.RESPONSIBILITY
          , a.RESPONSIBILITY_NAME
          , a.EXTRACT_DATE
    FROM temp_organogram a
    WHERE COMPONENT = j.COMPONENT;

    COMMIT;


END LOOP;
```

Appendix C

```
CREATE TABLE FLAT_ORGANOGRAM_DIM AS
SELECT
  to_char(a.EXTRACT_DATE,'dd-MON-yyyy')||'-'||a.COMPONENT          OGANOGRAM_SURROGATE_KEY,
  a.EXTRACT_DATE,
  a.STATUS,
  a.COMPONENT,
  a.SCBP,
  CASE WHEN (a.EFFECTIVE_DATE_FROM IS NOT NULL)  THEN
    TRIM(a.COMPONENT_NAME) || ' ('||a.RESPONSIBILITY||')' || ' ['||a.COMPONENT||'] '||
    ' ('||to_char(a.EFFECTIVE_DATE_FROM,'dd-MON-yyyy')||') - '||
    '('||to_char(a.EFFECTIVE_DATE_TO,'dd-MON-yyyy')||')'
  ELSE
    TRIM(a.COMPONENT_NAME) || ' ('||a.RESPONSIBILITY||')' || ' ['||a.COMPONENT||'] '
  END                                                              AS COMPONENT_NAME,
  TRIM(a.COMPONENT_NAME) || ' ['||a.COMPONENT||']'                 AS COMPONENT_NAME_CLEAN,
  a.COMPONENT_NAME                                                 AS COMPONENT_NAME_SHORT,
  a.COMPONENT_LEVEL                                                AS COMPONENT_LEVEL,
  a.CONTROL_COMPONENT                                              AS CONTROL_COMPONENT,
  a.EFFECTIVE_DATE_FROM                                            AS EFFECTIVE_DATE_FROM,
  a.EFFECTIVE_DATE_TO                                              AS EFFECTIVE_DATE_TO,
  substr(a.SCBP,2,6)                                               AS LEVEL1,
  substr(a.SCBP,9,6)                                               AS LEVEL2,
  substr(a.SCBP,16,6)                                              AS LEVEL3,
  substr(a.SCBP,23,6)                                              AS LEVEL4,
  substr(a.SCBP,30,6)                                              AS LEVEL5,
  substr(a.SCBP,37,6)                                              AS LEVEL6,
  substr(a.SCBP,44,6)                                              AS LEVEL7,
  substr(a.SCBP,51,6)                                              AS LEVEL8,
  substr(a.SCBP,58,6)                                              AS LEVEL9,
  substr(a.SCBP,65,6)                                              AS LEVEL10,
  substr(a.SCBP,72,6)                                              AS LEVEL11,
  substr(a.SCBP,79,6)                                              AS LEVEL12,
  substr(a.SCBP,86,6)                                              AS LEVEL13,
  substr(a.SCBP,93,6)                                              AS LEVEL14,
  substr(a.SCBP,100,6)                                             AS LEVEL15
FROM ORGANOGRAM a;
```

Appendix D

```
CREATE TABLE FLOW_EVENTS
(
  ADMISSION_KEY                 NUMBER(10),
  PATIENT_KEY                   NUMBER(10),
  FACILITY_KEY                  NUMBER(10),
  VISIT_DIM_KEY                 NUMBER(10),
  ACCOUNTNO                     VARCHAR2(50),
  REVENUE_SURROGATE_KEY         VARCHAR2(10),
  EVENT_DATE                    DATE,
  …
)
PARTITION BY LIST (FACILITY_KEY)
(
PARTITION FLOW_EVENTS_702           VALUES ('702'),
PARTITION FLOW_EVENTS_701           VALUES ('701'),
…
…
 PARTITION FLOW_EVENTS_411          VALUES ('411'),
 PARTITION FLOW_EVENTS_NA           VALUES (DEFAULT)
)
 NOLOGGING;
```

Appendix E

Part 1

SQL Code to Generate PatientData_Pivot

```
CREATE TABLE WH_PASIENTDATA_PIVOT
AS
SELECT
     Id              AS PASIENTDATA_ID,
     OrgUnitId       AS PASIENTDATA_ORGUNIT_ID,
     max(decode(DataElementId,3,  UPPER(TRIM(EntryText)))) SURNAME,
     max(decode(DataElementId,4,  UPPER(TRIM(EntryText)))) FIRSTNAME,
     max(decode(DataElementId,5,  UPPER(TRIM(EntryText)))) RES_ADDR,
     max(decode(DataElementId,6,  UPPER(TRIM(EntryText)))) POSTAREA,
     max(decode(DataElementId,7,  UPPER(TRIM(EntryText)))) DATEBRTH,
     max(decode(DataElementId,8,  UPPER(TRIM(EntryText)))) AGEYEARS,
     max(decode(DataElementId,9,  UPPER(TRIM(EntryText)))) AGEMONTH,
     max(decode(DataElementId,10, UPPER(TRIM(EntryText)))) GENDER,
     max(decode(DataElementId,127,UPPER(TRIM(EntryText)))) RACE,
     max(decode(DataElementId,11, UPPER(TRIM(EntryText)))) RURURBAN,
     max(decode(DataElementId,128,UPPER(TRIM(EntryText)))) PATSTATS
FROM
    (SELECT
        Id,
        OrgUnitId,
        DataElementId,
        EntryText,
        ROW_NUMBER() OVER (partition by Id,OrgUnitId order by DataElementId)
     FROM PatientData)
GROUP BY Id, OrgUnitId;
```

Part 2

SQL Code to Generate NotifData_Pivot

```
CREATE TABLE WH_NOTIFDATA_PIVOT
AS
SELECT
     Id                      AS PASIENTDATA_ID,
     OrgUnitId               AS PASIENTDATA_ORGUNIT_ID,
     max(decode(a.DataElementId,1,  UPPER(TRIM(EntryText))))         DISEASE,
     '00000000000000000000000000000000000000000000000000'  DISEASE_DECODED,
     max(decode(a.DataElementId,14, UPPER(TRIM(EntryText))))         DATE_IDENTIFIED,
     max(decode(a.DataElementId,70, UPPER(TRIM(EntryText))))    MEDICAL_DIAGNOSES,
     max(decode(a.DataElementId,130,UPPER(TRIM(EntryText))))    BASED_DIAGNOSES,
     max(decode(a.DataElementId,56, UPPER(TRIM(EntryText))))    DATE_DEATH,
     max(decode(a.DataElementId,61, UPPER(TRIM(EntryText))))    SURVTYPE,
     max(decode(a.DataElementId,131, UPPER(TRIM(EntryText))))  LAB_TEST1_TYPE,
     max(decode(a.DataElementId,132, UPPER(TRIM(EntryText))))  LAB_TEST1_RESULT,
     max(decode(a.DataElementId,133, UPPER(TRIM(EntryText))))  LAB_TEST2_TYPE,
     max(decode(a.DataElementId,134, UPPER(TRIM(EntryText))))  LAB_TEST2_RESULT,
     max(decode(a.DataElementId,135, UPPER(TRIM(EntryText))))  LAB_TEST3_TYPE,
     max(decode(a.DataElementId,136, UPPER(TRIM(EntryText))))  LAB_TEST3_RESULT
FROM
    (SELECT
        Id,
        OrgUnitId,
        DataElementId,
        EntryText,
        ROW_NUMBER() OVER (partition by Id,OrgUnitId order by DataElementId)
     FROM PatientData) a
GROUP BY Id, OrgUnitId
ORDER BY Id, OrgUnitId
```

Appendix F

Free State Department of Health – Data Warehouse Evaluation Questionnaire

Purpose: The purpose of this questionnaire is to assist the Free State Department of Health (FSDOH) in evaluating the current data warehouse with all its components of information already available. The output from the questionnaire will be used to improve our understanding of shortcomings in the data warehouse, and how it can be addressed to make it more user-friendly. **Please note that the questionnaire can be completed anonymously and that the contents of this questionnaire will be treated in a very confidential manner.**

Please circle the appropriate number of each question which is applicable to you and remember to answer it from your OWN point of view.

Record Number: _____

1. Gender

Male	1
Female	2

2. My Job level at the FSDOH

Production Staff – Level 1- 7	1
Supervisor – Level 8	2
Assistant Manager – Level 9-10	3
Manager – Level 11-12	4
Senior Manager – Level 13 +	5

3. Education level

Senior Certificate	1
1 year diploma	2
2 year diploma	3
3 year diploma or degree	4
4 year B.Tech or Honours degree or higher	5

4. My computer literacy level

Fairly new to a computer	1
Can help myself with basic tasks	2
Can help myself with intermediate tasks	3
Consider myself an expert	4

5. My computer experience level

Less than 6 months	1
Between 7 and 12 months	2
Between 13 and 24 months	3
More than 24 months	4

6. My computer analytical experience level (for example using Microsoft Excel)

Less than 6 months	1
Between 7 and 12 months	2
Between 13 and 24 months	3
More than 24 months	4

I. Use of Existing Data Warehouse					
5. How frequently do you access the data warehouse?	Daily	Weekly	Monthly	Quarterly	Never/almost never
	5	4	3	2	1
6. If you access the data warehouse *during the course of your own work*, what are the tasks and the frequency of access?	Very frequently	Frequently	Average	Rarely	Never
1. Status monitoring	5	4	3	2	1
2. Decision-making support	5	4	3	2	1
3. Forecasting	5	4	3	2	1
4. Administration	5	4	3	2	1
5. Accounting	5	4	3	2	1
6. Planning	5	4	3	2	1
7. Resource allocation/budgeting	5	4	3	2	1
8. Personnel management	5	4	3	2	1
9. Parliamentary enquiries	5	4	3	2	1

3. How do you extract or request reports from the data warehouse?		
	By myself	1
	From an assistant	2
	From knowledge workers at Head Office	3

7. With which frequency would you say do you access the data warehouse for the following:	Daily	Weekly	Monthly	Quarterly	Never/almost never
1. To create reports for supervisors or managers	5	4	3	2	1
2. To acquire information for my own work	5	4	3	2	1
3. For external customer services	5	4	3	2	1
4. To answer parliamentary information requests	5	4	3	2	1
5. To answer research questions	5	4	3	2	1
6. To acquire information for monthly or quarterly performance or progress reports	5	4	3	2	1

5. With which frequency do you access the following parts of the data warehouse?	Very frequently	Frequently	Average	Rarely	Never
Human Resources	5	4	3	2	1
PADS2 (revenue collection, patient statistics)	5	4	3	2	1
Antiretroviral Therapy	5	4	3	2	1
Notifiable Diseases	5	4	3	2	1
Tuberculosis (TB)	5	4	3	2	1
II. Perception of Information from the Data Warehouse					
1. To what degree would you agree with the following statements	Agree strongly	Agree	Unsure	Disagree	Disagree strongly
1. I can get data that is current enough to meet my work needs	5	4	3	2	1
2. The data warehouse maintains data at an appropriate level of detail for my tasks	5	4	3	2	1
3. The data in the data warehouse is accurate and reliable	5	4	3	2	1
4. There are too many features in the data warehouse that have no functional value to me	5	4	3	2	1
5. It is easy to locate data on a particular issue in the data warehouse, even if I haven't used that data before	5	4	3	2	1
6. The data in the data warehouse is out of date and is not addressing my work needs	5	4	3	2	1
7. I am satisfied with the overall quality of information generated from the data warehouse	5	4	3	2	1
8. I am satisfied with the overall performance of the data warehouse system in its *functionality*	5	4	3	2	1
9. I am satisfied with the overall performance of the data warehouse system in its *flexibility*	5	4	3	2	1
10. I am satisfied with the overall performance of the data warehouse system in its *processing speed*	5	4	3	2	1
11. I find it difficult to look for relevant information in the data warehouse to answer a specific business question	5	4	3	2	1
12. There are times when I find that supposedly equivalent data is inconsistent in the data warehouse	5	4	3	2	1
13. The data warehouse system is convenient and easy to use	5	4	3	2	1
14. I would like more training for me or my staff on how to find, understand, access or use the data warehouse	5	4	3	2	1
15. I sometimes have trouble accessing the data warehouse and am forced to wait	5	4	3	2	1
16. The information from the data warehouse does not have enough detail to make me more productive	5	4	3	2	1

17. The information from the data warehouse lacks quality and more control measures must be put in place	5	4	3	2	1
18. The data warehouse does not allow me to perform my own analysis and draw my own conclusions	5	4	3	2	1
19. Overall I find the data warehouse to be a valuable asset for the FSDOH	5	4	3	2	1

III. Suggestions

Are there any suggestions you'd like to make on how the data warehouse can be improved?

1. ……

2. ……

3. ……

You can email the completed questionnaire to KotzeE@fshealth.gov.za or alternatively fax it to 051 408 1913

Thank you for your time in completing this questionnaire!

Appendix G

The Medical Research Council

Biomedical Informatics Research Division

PO Box 19070, Tygerberg 7505, Cape Town, South Africa
Francie van Zijl Drive, Parow Valley, Cape Town
Tel +27 21 938 0851, Fax +27 21 938 0526

14 November 2008

Mr Eduan Kotze
Principal Data Technologist
Free State Department of Health
Bloemfontein

Dear Eduan

LONGITUDINAL PATIENT TRECORD SYSTEM

Thank you very much for developing the Longitudinal Patient Record (LPR) application and making it available to us. This is a major achievement and a substantial help to our efforts to strengthen service delivery for patients accessing the Free State public sector antiretroviral treatment program.

There are several features of the system that are particularly useful for us:

1) The web accessible front end is very useful as it means that we are able to access data from any site with Internet access.
2) The structuring of data in terms of events, ie an encounter-based system is extremely important as it enables us to compile longitudinal; fdata sets that can be used for cohort analyses.
3) The integration of longitudinal patient data from patients registered on the HIV/antiretroviral treatment program with other data sets, including the provincial TB Control Program and the vital registration system in one accessible record is extremely useful and, I believe, a first in South Africa.

We have recently launched a research program to implement treatment failure clinics in the Free State and hope to work with you further to build consistent data sets for patients failing treatment and to analyze the data to provide new insights into causes and mechanisms of treatment failure. This data resource is unsurpassed in terms of its ability to support this kind of analysis. We also hope to develop a patient failure summary sheet on top of this data.

Congratulations on this achievement and I look forward to working with you going forward.

Yours sincerely

Dr Chris Seebregts
SENIOR MANAGER

SUMMARY

The AIDS epidemic, caused by Human Immunodeficiency Virus (HIV), is a global crisis which threatens development gains, economies, and societies. The eradication of HIV/AIDS represents one of humanity's greatest challenges, which requires co-operation, and comprehensive collaboration between science, governments, social institutions, the media, the professions, and the general public. In this endeavour strategic information plays a major role.

The Free State Department of Health introduced Antiretroviral Therapy (ART) to meet the needs of the rising number of people living with HIV and also dying due to AIDS related diseases. A patient information system was deployed by the Province to supplement the ART programme rollout process by gathering data and providing all the basic patient antiretroviral information. The patient information system was a traditional online clinical system and concentrated on the bread-and-butter issues of accumulating data on a patient. Very little functionality was provided to deal with the complexities of managing the clinical outcomes of the ART programme. To add to the problem, other online operational systems had to be interrogated to gain an understanding of the impact the rollout of ARV had. These operational systems ranged from standalone Human Resource systems to information systems accumulating data on tuberculosis which is closely related to HIV/AIDS. No mechanism or platform existed to provide management with integrated strategic information to manage the business process intelligently.

This study focussed on the challenges and solutions to overcome this shortfall. The main objective of this study was to construct a framework for providing strategic information for the management of the ART programme in the Free State Department of Health.

This objective was reached with a two phase *action research* methodology. The author of this study fulfilled simultaneously the roles of the *researcher* and the *practitioner*. During phase one a central data warehouse was designed and developed that incorporated an existing standalone ARV data warehouse and several other independent operational sources, all related to ART. This warehouse was evaluated by the users who overwhelmingly rated the warehouse as successful. From one system it was possible for managers to obtain strategic information on ARV encounters, ARV human resources, revenue collection, in-patient hospitalization, notifiable diseases and tuberculosis. This was achieved with a very limited budget and using internal resource.

Although the data warehouse was a valuable asset by itself, several improvements were identified during the evaluation of the warehouse. The most important of these were to add additional data marts, to conform all the dimensions in order to obtain strategic information across the data marts and lastly to be able to track an ARV patient over time over all facilities of care. This led to the second phase of the research. In this phase several data marts (National Health Laboratory Services (NHLS), Meditech hospitalization and linkage to Home Affairs) were firstly added. Secondly the patient dimensions of all the data marts were conformed through the process of probabilistic record linkage. Lastly a longitudinal patient record was developed that displayed all the encounters of an ARV patient over time. Even though the LPR could not be scientifically evaluated, the institutions that did test it rated it very highly and reckoned it was a first for South Africa.

This study, therefore, concluded with the fact that it was possible to construct and implement a successful framework for providing comprehensive strategic information for the management of the ART programme in the Free State Department of Health.

Key words:
Data warehousing
Data marts
Dimensional modeling
Business intelligence
Antiretroviral therapy
Healthcare
Record linkage
Probabilistic matching
Longitudinal patient record

OPSOMMING

Die VIGS epidemie wat deur die menslike-immuniteitsgebrekvirus (MIV) veroorsaak word, is ‘n globale krisis wat ontwikkelingsvooruitgang, ekonomië en gemeenskappe bedreig. Die vernietiging van MIV/VIGS is een van die mensdom se grootste uitdagings en dit vereis koördinasie en omvattende medewerking tussen die wetenskap, staatsbestuur, maatskaplike instellings, die media, die professies en die algemene publiek. In hierdie strewe speel strategiese inligting ‘n belangrike rol.

Die Vrystaatse Departement van Gesondheid het Antiretrovirale Terapie (ART) ingestel om die behoeftes aan te spreek van die stygende getal mense wat leef met MIV en ook sterf weens VIGS. ’n Pasiëntinligtingstelsel was deur die Provinsie ontplooi om die ART implementeringsprogram aan te vul deur die nodige data te versamel en die basiese pasiënt antiretrovirale inligting te voorsien. Die pasiëntinligtingstelsel was ’n tradisionele gekoppelde kliniese stelsel, wat gekonsentreer het op die brood-en-botter sake van die versameling van pasiëntdata. Baie min funksionaliteit was beskikbaar om die kompleksiteit van die kliniese uitslag in die ART program te bestuur. Bydraend tot die probleem, moes navrae ook gedoen word op ander onafhanklike operasionele stelsels ten einde begrip te verkry van die impak wat die implementering van ART gehad het. Hierdie operasionele stelsels het gewissel van alleenstaande menselike hulpbron stelsels tot inligtingstelsels wat data versamel het oor tuberkulose, wat nou met MIV/VIGS verwant is. Geen meganisme of beleid het voorheen bestaan om geïntegreerde strategiese inligting aan bestuur te voorsien om die besigheidsproses optimaal te bestuur nie.

Hierdie studie het gevolglik gefokus op die uitdagings en oplossings om die bogenoemde tekortkominge te bowe te kom. Die hoof oogmerk van hierdie studie was om ’n raamwerk vir die voorsiening van strategiese inligting vir die bestuur van die ART program in die Vrystaatse Departement van Gesondheid te ontwikkel en in werking te stel.

Hierdie doelwit is deur ’n twee-fase *aksienavoringsmetodologie* bereik. Die skrywer van hierdie studie het beide die rolle van *navorser* en *praktisyn* gelyktydig vervul. Gedurende fase een was ’n sentrale datapakhuis ontwerp en ontwikkel wat ’n bestaande alleenstaande ART datapakhuis geïnkorporeer het asook ander onafhanklike operasionele bronne wat aan ART verwant is. Hierdie datapakhuis was deur gebruikers ge-evalueer en as oorweldigend suksesvol bestempel. Dit was nou moontlik vir bestuurders om vanuit een stelsel die nodige strategiese inligting oor ART besoeke, ART menslike hulpbronne, inkomsteversameling, binnepasiënt hospitalisasie, aanmeldbare siektes en tuberkulose te bekom. Hierdie doel was bereik deur ’n beperkte begroting en die benutting van interne hulpbronne.

Ondanks die feit dat die datapakhuis in eie reg ’n waarvolle bate is, is verskeie verbeterings gedurende die evaluasie van die datapakhuis geïdentifiseer. Die belangrikste hiervan was die toevoeging van addisionele datamarte, om dimensies te konformeer en strategiese inligting dwarsoor die datamarte te voorsien om sodoende ’n ART pasient te kon naspoor oor tyd en alle fasiliteite van sorg. Dit het aanleiding gegee tot die tweede fase van die navorsing. In hierdie fase was verskeie datamarte (National Health Laboratory Services (NHLS), Meditech hospitalisasie en die skakeling met Binnelandse Sake) eers toegevoeg. Tweedens was die pasiëntdimensie in alle datamarte gekonformeer deur die proses van waarskynlikheids-rekordskakeling. ‘n Longitudinale pasiëntrekord (LPR) is daarna ontwikkel, wat alle kontak met ‘n ART pasiënt oor ‘n tydperk weerspieël het. Alhoewel die LPR nie wetenskaplik ge-evalueer was nie, het instansies wat dit wel getoets het, dit baie hoog aangeskryf en die algemene mening was dat dit ‘n eerste vir Suid-Afrika is.

Die studie sluit af met die gevolgtrekking dat dit moontlik was om ’n raamwerk vir die voorsiening van strategiese inligting vir die bestuur van die ART program in die Vrystaatse Departement van Gesondheid suksesvol te ontwikkel en in werking te stel.

Sleutelwoorde:
Datapakhuis
Datamarte
Dimensionêle modelering
Besigheidsintelligensie
Antiretrovirale terapie
Gesondheidsorg
Rekord skakeling
Waarskynlikheids soortgelyk
Longitudinale pasiëntrekord

Managing Worldwide Operations and Communications with Information Technology

2007 Information Resources Management Association International Conference
Vancouver, British Columbia, Canada May 19-23, 2007

Mehdi Khosrow-Pour
Information Resources Management Association, USA

IGI PUBLISHING
Hershey • New York
http://www.igi-pub.com

Challenges in developing a data warehouse to manage the rollout of antiretroviral therapy in a developing country

J.E. Kotzé & T. McDonald
Department of Computer Science and Informatics,
University of the Free State, South Africa

Abstract

With a global HIV/AIDS epidemic, developing countries are facing an enormous challenge in combating the disease. Public health will be placed under severe pressure in providing treatment such as highly active antiretroviral therapy to all its HIV infected patients. This paper will describe the challenges involved in establishing a data warehouse to provide strategic information during the rollout of antiretroviral therapy (ART). The construction of a Human Resources Data Mart, which is critical to the successful rollout of antiretroviral therapy in South Africa, will be discussed in detail. Special attention will be given to extraction, transformation and loading, slowly changing dimensions type 2 and materialized views.

Keywords: Data Warehousing, Healthcare, Human Resource Data Mart, Antiretroviral Treatment, Developing Countries.

1. Introduction

The HIV/AIDS epidemic is a global crisis which threatens development gains, economies and societies. At the end of 2004, the total number of people worldwide living with HIV/AIDS was estimated to be just under 40 million. In South Africa the estimated number of AIDS related deaths in 2003 ranged anywhere between 270 000 and 520 000 according to the UNAIDS Global Report (UNAIDS, 2004).

In response to this epidemic, the South African Government created the HIV/AIDS and STD Strategic Plan. This plan includes the provision of antiretroviral therapy in the public health sector in an attempt to reduce AIDS mortalities. Antiretroviral treatment (ART) for HIV infection consists of drugs that slow down the reproduction of the HIV virus in the body.

The Free State Department of Health (FSDOH) launched its provincial antiretroviral treatment program during May 2004. By the end of June 2006 a total of 31 public health facilities were empowered to provide antiretroviral drugs for 6200 patients in the Free State. The Actuarial Society of South Africa (ASSA) has developed an AIDS Demographic Model that can be used to project the impact of this disease on each province in South Africa. According to Chapman (2003) using the ASSA 2000 Model, it is estimated that in the Free State

- Approximately 480 000 people are HIV positive (based on 30.1% HIV positive mothers in the 2003 HIV Antenatal Survey);
- Seven percent (7%) of all HIV infected patients are in a World Health Organization (WHO) Stage 4 AIDS defining illness, which is approximately 31 111 patients;
- Annually, 28 290 patients will develop a WHO Stage 4 AIDS defining illness.

The WHO recommends that all people in a WHO stage 4 AIDS defining illness should commence with antiretroviral treatment immediately. This recommendation will pose serious challenges in managing the resources required for treating all these patients by the FSDOH. Mechanisms have to be developed to effectively monitor the antiretroviral treatment programme but at the same time provide the necessary **strategic**

information in managing and evaluating the programme as well. It is clear that a number of factors are forcing the FSDOH in the direction of a data warehouse (DW).

This paper will indicate how a Health Department in South Africa, the FSDOH, tackled and successfully managed the challenges of creating a data warehouse. A background section will provide the history of the current operational system and the shortcomings of the system. That will be followed by a detailed discussion of the challenges involved in constructing the human resource data mart (HRDM) which is critical to the successful rollout of ART in South Africa. The Extraction, Transformation and Loading (ETL) process will be examined and slowly changing dimensions will be addressed. The paper then concludes with a discussion of a modified staging area that uses a materialized views approach to provide the platform for developing the human resource (HR) online analytical processing (OLAP) cube.

2. Background

2.1 General

A data warehouse differs significantly from a conventional operational or transactional database in several aspects. First of all, a complex data structure must be maintained in order to offer flexible and dynamic retrieval of rich decision-support knowledge (Shin, 2003). For this, it maintains data that is more integrated, subject-oriented, non-volatile and time-variant in comparison with transactional or operational databases (Dodge & Gorman, 2000; Hristovski *et. al.,* 2000; Shin, 2003). Data structures of a data warehouse should also be more cross-functional (Shin, 2003) and support management decisions (Hristovski *et. al.,* 2000).

According to Saraceni *et. al.,* (2005), the linkage of several databases can assist with studying the distribution of diseases and for analysis of AIDS-related mortalities in Brazil. Although the linkage of databases is in essence not a data warehouse, it demonstrates the importance of analyzing information and using it to provide strategic information for the decision-making process.

Data warehouses have previously been used in the areas of health and public health (Davis *et. al.*, 2002; Lau & Catchpole, 2001; Prather *et. al.,* 1997). However, most data warehouses in the health areas are used for ***clinical treatment outcome*** or for ***biomedical studies*** and limited research has been done on the usage of data warehouses in public health for holistic decision-making.

2.2 Lack of strategic information

The Personnel and Salary (PERSAL) system is an online transaction processing (OLTP) based payroll system and is used by all the National and Provincial governments in South Africa. The system has been in a production environment since 1990 and was developed in Natural Adabas. At present, the system is being maintained by a private company.

Because the system is OLTP based, it proved inadequate in providing the necessary human resource statistics needed by antiretroviral programme managers. Furthermore, every new change or new report must be submitted to a central *System Change Control* system. From there it will be prioritised and once accepted, handed over to the private company for development.

This process was cumbersome, inflexible and time consuming which led to overall frustration. In 2004, the FSDOH received approval from National Treasury to extract all relevant human resource data from PERSAL, thus allowing them the freedom of incorporating the data into a data warehouse.

3 Challenges

The following sections will provide details on the challenges that were faced during the development of a data warehouse.

3.1 Limited budget

According to Schubart & Einbinder (2000), research has showed that the key factors for successful data warehouse implementation are organizational in nature. Management support and adequate resources are most important because these address political resistance. Gatziu & Vavouras (1999) stated that data warehouse development is a demanding and costly activity of which the establishment thereof could be in excess of $1m.

This can be a major obstacle in a developing country.

Taking these factors into consideration, top management was approached to direct the development of the data warehouse in early 2005. Because of a limited IT budget (0.68%), a decision was taken to break the project down into several data marts and to develop the data warehouse over a longer period of time. To cut back on costs, in-house existing staff was used to construct the data warehouse in lieu of making use of expensive outside consultancy firms.

Oracle is the current worldwide leader in the data warehouse tools marketplace (Vesset, 2006). Furthermore, Oracle 10.2g also offers all the functionalities required in both OLTP and DW based databases. Both these reasons guided the FSDOH decision to upgrade their existing Oracle9i infrastructure to Oracle 10.2g and make use of it for the data warehouse. The upgrade process was covered in an existing maintenance contract, resulting in no additional expenditure.

Human resources and pharmaceutical (ART drugs) costs were identified as the main cost drivers, and more strategic information was required in order to obtain sufficient funding for the ART programme.

Listed below are the identified data marts:

- human resources
- clinical patient ART treatment
- pharmaceuticals (ART drugs)
- patient mortalities
- tuberculosis

The Human Resource Data Mart (HRDM) was chosen as the first data mart to be constructed and will be the focus of the rest of this paper.

3.2 Extraction, Transformation and Loading Challenges

3.2.1 Data Extraction

The data in the OLTP system (PERSAL) is *transient data* of nature. According to Bruckner & Tjoa (2002), the key characteristic of *transient data* is that alterations and deletions of existing records physically destroy the previous data content. In order to keep the history of the data in tact, all modifications to the data had to be considered.

The ETL processes commenced with a data extraction process which are performed **twice** a month by Treasury. At the beginning of every month, the FSDOH will receive two sets of data. The reason for this approach is entrenched in the manner in which government officials receive their salaries in South Africa. The salary of permanent staff is paid on the 15th of each month. All the information related to this event constitutes data set one. However, additional or supplementary payments (i.e. S&T claims, overtime, fuel allowance) and workforce operations (promotions, staff re-allocations) can be made to government officials from the 16th until the end of the month. All these additional information constitutes data set two.

In essence, the first extraction consists of a full *data snapshot* taken on the 15th of the month from the *transient data set*. This includes **staff, posts** and the **hierarchical organization structure**. According to Bruckner & Tjoa (2002), a *data snapshot* is a stable view of data as it exists at some point in time. It is a special kind of periodic data. Snapshots usually represent the data at some time in the past, and a series of snapshots can provide a view of the history of an organization.

The second extraction consists of the supplementary changes to the *data snapshot* picture of the first extraction set in terms of **staff** and **posts** but **exclude** any changes to the hierarchical organization structure. This extraction process was preformed at the end of the month. It can be regarded as *semi-periodic data*. According to Bruckner & Tjoa (2002), almost all operational systems retain only a small history of data changes due to performance and/or storage constraints.

The challenge pertaining to more than one set of extraction data in the update window is the issue of *late-arriving data*. According to Bruckner & Tjoa (2002), *late-arriving data* is bothersome because it is difficult to

integrate with existing fact and dimension tables, especially when surrogate keys are used in order to cope with slowly changing dimensions. Aggregates have to be updated, because the newly integrated data sets will change counts and totals of the prior history. Late-arriving data can therefore possibly change analysis results unexpectedly from the analyst's perspective.

In order to deal with the problem of late-arriving data, it was agreed that the data warehouse will be updated during the **first week** of the following month, reflecting the *transient data picture* and supplementary changes (*semi-periodic data*) that was made to it.

3.2.2 Time Stamping

The standard approach for storing periodic data (typically found in Data warehouses) is to use time stamped status fields for each record. For the HRDM the *load timestamp* method was used.

Slow changing dimensions (SCD) Type 2 will be used as far as possible. According to Berndt & Fisher (2001), this type of change adds rows to maintain an arbitrarily long history. The keys must be "generalized" in this approach by using a version number or some other mechanism, so that related rows can be retrieved as a coherent history.

Each table in the staging area had a column added called EXTRACT_DATE which translated to the record *load timestamp.*

3.2.3 Dealing with Slowly Changing Dimensions

One of the biggest challenges with the HRDM was the monthly changes to the organizational structure. Changes occur when new components (organizational units) are created, moved or become obsolete during the month. Components contain the posts for that particular unit and the links of the child components directly reporting to it.

Changes in the organizational structure were not directly reflected in each month's download and had to be identified with specially developed algorithms in order to perform SCD Type 2. This was because the organizational structure was only included in extraction set 1 as a *data snapshot picture* called **Organogram.txt** and not a list of changes. See Figure 1 for the organizational structure data flow of June 2005 as an example.

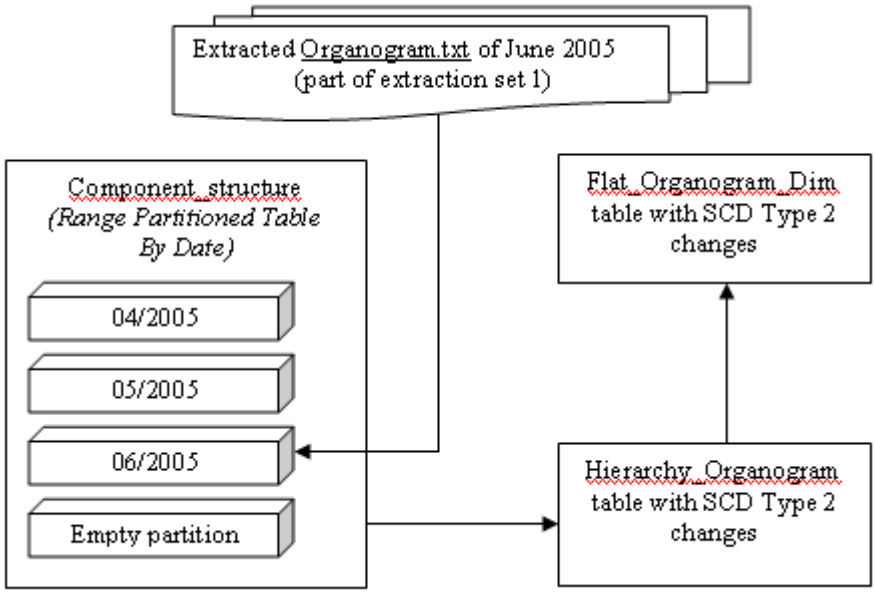


Figure 1: Data Flow for Extracted Organogram.txt (June 2005)

The data for the organizational structure was imported into the COMPONENT_STRUCTURE partitioned table from the Organogram.txt file. This partitioned table then contained the organizational hierarchy for each month. The organizational hierarchy in turn, consisted of component details and linkages between child and parent components.

A table called HIERARCHY_ORGANOGRAM was constructed and populated with the hierarchy on the date the HRDM project commenced (April 2005). For each following month, COMPONENT_STRUCTURE was algorithmically compared to HIERARCHY_ORGANOGRAM using complex SQL statements and SET operators to help identify the following changes:

- New Component
- Component name change
- Parent component position change
- Parent component name change
- Deleted Component

Each time a change was detected, a new record was inserted into HIERARCHY_ORGANOGRAM, with a new surrogate key. The superseding record was changed to the last date of the previous month. A surrogate key called ORGANOGRAM_KEY was created with the extract date (or *load timestamp*) concatenated with the component number to stay within the bounds of SCD Type 2.

Kimball & Margy (2002) pointed out that hierarchical structures of variable depth presents several problems in the relational environment. Some examples are the difficulty of navigation or rolling up of facts within these hierarchies using non-procedural SQL. This posed a problem for the FSDOH when using Oracle 'CONNECT BY' SQL extension in the same statement as a join. While 'CONNECT BY' is very useful when navigating recursive points in a dimension table, it can not be used by an ad hoc query tool. If the tool could generate this syntax to explore the recursive relationship, it cannot in the same statement be joined to a fact table. Even if Oracle was to remove this somewhat arbitrary limitation, the performance at query time would probably be not too good (Corr, 2001).

To overcome this problem, a bridge table or often called helper tables are inserted between the hierarchical dimension table and the fact table (Kimball & Margy, 2002). The problem the FSDOH experienced with this approach was entrenched in the manner the multidimensional online analytical processing (MOLAP) tool used the dimensional model for its analytical model. The MOLAP tool required a flat organizational view which in theory meant a totally denormalized view of the hierarchical organizational structure and relationships in HIERARCHY_ORGANOGRAM.

Kimball & Margy (2002) also pointed out that when navigating the bridge table via the standard SQL code, it is not for the faint of heart. In order to overcome the prerequisite of the MOLAP tool together with minimizing the SQL complexity for the FSDOH users, a ***modified version*** of a bridge table was introduced. The table FLAT_ORGANOGRAM_DIM was created and used as one of the dimensions in the dimensional model (See Figure 2).

This ***modified version*** of a bridge table might not be the perfect solution should the organizational structure consist of more than 10 levels. To overcome this, the table is re-created every month from all the SCD Type 2 changes captured within HIERARCHY_ORGANOGRAM. In this manner the algorithm will allow an extra level (meaning an extra table column) when it detects it, thus avoiding the possibility of missing data. However, the only manual action to be taken is to insert this additional level (table column) within the MOLAP tool.

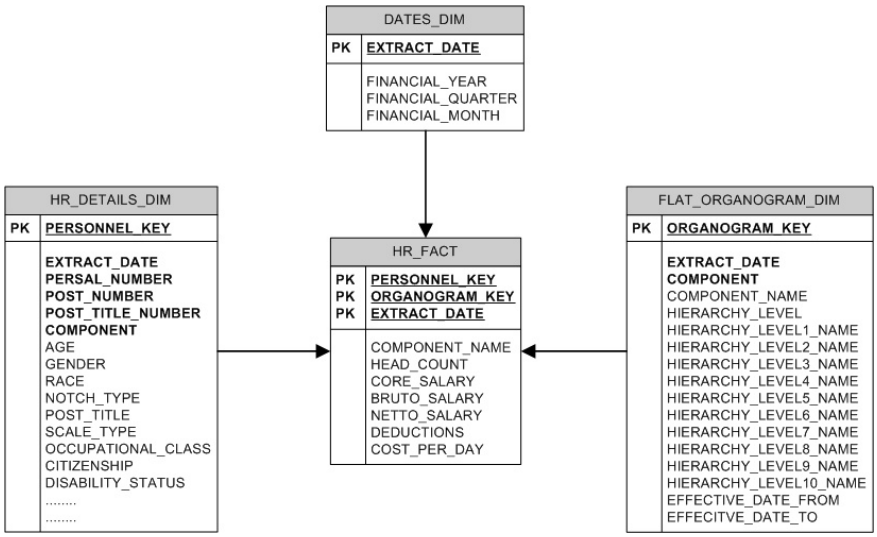


Figure 2: Dimensional Model

3.2.4 Example of a SCD Type 2 on a Parent Component

The following example (See Figure 3) will illustrate a SCD Type 2 on a parent Organizational Unit (Component) between April 2005 and June 2005 and the domino effect it will have on its child components.

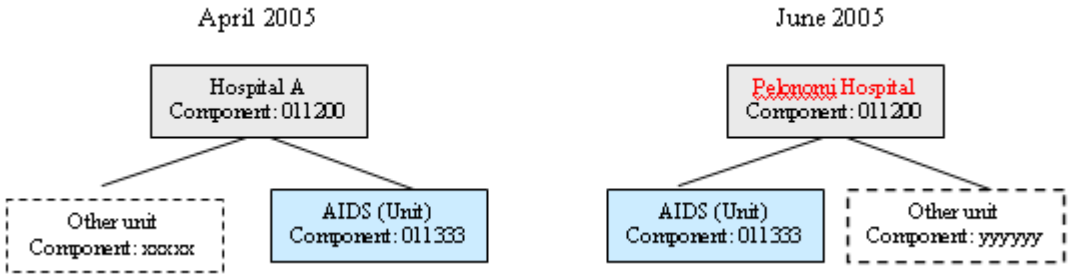


Figure 3: Parent Component Name change

- Step 1: Algorithm detects a change in *parent name* in component 011200
- Step 2: Perform SCD Type 2 and load changes into HIERARCHY_ORGANOGRAM

HIERARCHY_ORGANOGRAM					
Organogram Key	Extract date	Component Name	Parent Component	Date From	Date To
01-APR-2005-011200	01-APR-2005	Hospital A	011000	01/04/2005	30/06/2006
01-JUL-2006-011200	01-JUL-2006	Pelonomi Hospital	011000	01/06/2006	

- Step 3: Force the change in all child components of component 011200. Only ***Component 011333*** will be illustrated below.

HIERARCHY_ORGANOGRAM					
Organogram Key	Extract date	Component Name	Parent Component	Date From	Date To

01-APR-2005-011333	01-APR-2005	AIDS (Unit)	011200	01/04/2005	30/06/2006
01-JUL-2006-011333	01-JUL-2006	AIDS (Unit)	011200	01/06/2006	

- Step 4: Convert the hierarchical organizational structure into a flat organizational structure

FLAT_ORGANOGRAM_DIM							
Organogram Key	Extract date	Component Name	Effective Date From	Effective Date To	Level 1	…	Level x
01-APR-2005-011200	01-APR-2005	Hospital A	01/04/2005	30/06/2006	HEALTH		Region A
01-JUL-2006-011200	01-JUL-2006	Pelonomi Hospital	01/06/2006		HEALTH		Region A
01-APR-2005-011333	01-APR-2005	AIDS (Unit)	01/04/2005	30/06/2006	HEALTH	….	Hospital A
01-JUL-2006-011333	01-JUL-2006	AIDS (Unit)	01/06/2006		HEALTH	….	Pelonomi Hospital

3.2.5 Using Materialized Views and SCD Type 2

According to Becker (2004), one of the problems of the SCD Type 2 technique is the large number of additional rows required to support all the changes. Barbusinski *et. al.*, (2003) pointed out that joining the fact and associated dimensions would also require complex temporal joins at analysis time. Furthermore the SQL statement must include time reference for both the fact and associated dimensions. All these factors will lead to an undesired environment for non-sophisticated users such as in the case of the FSDOH.

One possible way of overcoming these obstacles, is by using a materialized view (mview) to hide the complexity. A materialized view also physically stores the data that corresponds to the view's defined query (Dodge & Gorman, 2000). According to Goldstein & Larson (2001) query processing time can be improved through the use of materialized views.

For these reasons it was decided to make use of Oracle's materialized views. HR_DETAILS_DIM (mview) was created by joining all the posts with the matching staff member details. A staff member could also belong to more than one post. In order to uniquely identify a staff member with a particular post, a surrogate key called PERSONNEL_KEY was constructed for this purpose.

The PERSONNEL_KEY was constructed using a concatenated combination of the following fields from the posts and staff tables:

- EXTRACT_DATE (staff details)
- PERSAL_NUMBER (staff details)
- POST_NUMBER (post details)
- POST_TITLE (post details)
- COMPONENT (post details)

Thereafter, HR_FACT (mview) was created by joining FLAT_ORGANOGRAM_DIM (table) and HR_DETAILS_DIM (mview) to ensure consistency with all the SCD Type 2 changes in FLAT_ORGANOGRAM_DIM.

3.2.6 Building OLAP cubes

The HRDM OLAP cube was constructed from the dimensional model (See Figure 4). This was all done using Cognos Framework Manager and Cognos Transformer Series 7.

Research done by Gorla (2003) to evaluate OLAP tools in ease of use and usefulness, suggested that MOLAP be used for non-sophisticated computer users and relational online analytical processing (ROLAP) for sophisticated users.

Since most of the users at FSDOH can be categorized as non-sophisticated computer users, the **MOLAP** architecture was the choice of platform. The cube was deployed using Cognos Enterprise Server Series 7 which delivers **Web-based OLAP (WOLAP)** content, but using an underlying architecture that is still MOLAP. According to De Beer (2006), WOLAP is also seen as the next generation BI tool providing "thin-client" viewing tools for analyzing information.

The users were able to generate pivot tables (See Figure 4) from the WOLAP cube to assist them in obtaining strategic human resource information.

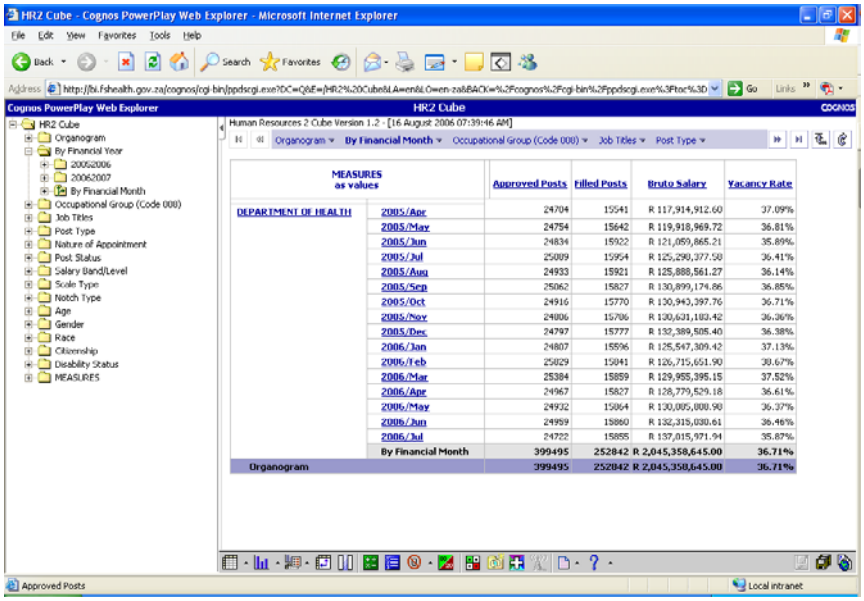


Figure 4: Pivot table from Human Resource data mart

4 Conclusion

Efficient resource management is critical for the success of the rollout of antiretroviral therapy in South Africa. Human resources management and ART drugs management was identified as the key factors but also the main cost drivers. A HRDM was built to provide strategic information for the ART programme. FSDOH management was now able to perform efficient staff allocations, monitor absenteeism and identify overworked personnel in time. Problematic ART clinics and hospitals in terms of staff turnaround could now also be easily identified by using trends, providing the FSDOH management team enough time to address the problem.

Future work and research could be done to link the HRDM to the ART clinical data set to identify health workers infected with HIV and AIDS. With this information, FSDOH management can obtain a better picture on the infection rate of HIV and AIDS on its health workers.

In conclusion, this paper demonstrated that it is possible to overcome the challenges of building a large-scale data warehouse, by starting small, using in-house knowledge and skills and to build data mart by data mart. The ETL process was modified to overcome the challenge of using SCD Type 2 within a hierarchical dimension. Materialized views were used to assist with the construction of the OLAP cube by camouflaging the complexities created by SCD Type 2. The end result was a MOLAP cube which provided an environment, conducive for analytical HR operations.

5 References

1. Barbusinski L, Howard S, Jennings M, Kelley C, Oates J. (2003). The relationship between a fact and dimension table, DMReview.com http://www.dmreview.com/article_sub.cfm?articleId=6349

2. Becker B. (2004). Kimball Design Tip #53: Dimensions Embellishments. Retrieved September 3, 2006, from http://www.rkimball.com/html/designtipsPDF/KimballDT53Dimension.pdf

3. Berndt DJ, Fisher JW. (2001). Understanding Dimension Volatility in Data Warehouses (or Bin There Done That). *Sixth INFORMS Conference on Information Systems and Technology* (INFORMS/CIST-2001)

4. Bruckner RM, Tjoa AM. (2002). Capturing Delays and Valid Times in Data Warehouses – Towards Timely Consistent Analyses. *Journal of Intelligent Information* Systems, 19(2), p169-190. Kluwer Academic Publishers.

5. Chapman RD. (2003). Plan for Implementation of ARV's in the Free State Province. Unpublished.

6. Corr L (2001). Kimball Design Tip #17: Populating Hierarchy Helper Tables. Retrieved September 22, 2006, from http://www.kimballgroup.com/html/designtipsPDF/DesignTips2001/KimballDT17Populating.pdf

7. Davis X, Wan C, Ross L, Wen X & Thomas B. (2002). A data warehouse concept for HIV prevention program evaluation. *AIDS Education and Prevention: Official Publication Of The International Society For AIDS Education*, 14(3 Suppl A), p120-122

8. De Beer, E (2006). Mobilising BI with data quality. Computing SA, July 2006, p35

9. Dodge G, Gorman T. (2000). Essential Oracle8i Data Warehousing: Designing, Building and Managing Oracle Data Warehouses. Wiley. New York.

10. Department of Health (Free State) (2006). Corporate Strategic Plan 2006/2007 to 2014/2015

11. Gatziu S, Vavouras A. (1999). Data warehousing: Concepts and mechanisms. INFORMATIK Volume 1 of 1999

12. Goldstein J, Larson P. (2001). Optimizing Queries using Materialized Views: A Practical Scalable Solution, ACM SIGMOD, 2001

13. Gorla N. (2003). Features to consider in a data warehousing system. *Communications of the ACM,* 46(11), p111-115

14. Hristovski D, Rogac M, Markota M. (2000). Using Data Warehousing and OLAP in Public Health Care. *Journal – American Medical Informatics Association*, 7, p369-373.

15. Kimball R, Margy R. (2002). The Data Warehouse Toolkit. Wiley Computer Publishing. Second Edition.

16. Lau RKW, Catchpole M. (2001). Improving data collection and information retrieval for monitoring sexual health. *International Journal of STD and AIDS*, 12(1), p8-13

17. Prather JC, Lobach DF, Goodwin LK, Hales JW, Hage ML, Hammond WE. (1997). Medical data mining: Knowledge discovery in a clinical data warehouse. *Proceedings of the 1997 AMIA Annual Fall Symposium, Bethesda: American Medical Informatics Association,* p101-105

18. Saraceni V, Da Cruz MM, Lauria LM, Durovni B. (2005). Trends and Characteristics of AIDS Mortality in the Rio de Janeiro City after the Introduction of Highly Active Antiretroviral Therapy. The Brazilian Journal of Infectious Diseases 2005, p209-215

19. Schubart J, Einbinder J. (2000). Evaluation of a data warehouse in an academic health sciences center. *International Journal of Medical Informatics*, p319-333

20. Shin B. (2003). An Exploratory Investigation of System Success Factors in Data Warehousing. *Journal of the Association for Information Systems,* Volume 4, p141-170.

21. UNAIDS.ORG. (2004). Report on Global AIDS epidemic, July 2004. Retrieved February 2, 2006, from http://www.unaids.org/bangkok2004/report_pdf.html

22. Vesset, D (2006). Worldwide Data Warehousing Tools 2005 Vendor Shares, IDC #203229, Volume 1. Retrieved December 1, 2006, from http://www.oracle.com/corporate/analyst/reports/infrastructure/bi_dw/idc-dw-tools-2005-1.pdf