

© Copyright 2013

Alicia F. Guidry



Ontology-Based Data Integration of Open Source Electronic Medical Record and  
Data Capture Systems

Alicia F. Guidry

A dissertation

Submitted in partial fulfillment of the  
Requirements for the degree of

Doctor of Philosophy

University of Washington

2013

Reading Committee:

James F. Brinkley III, Chair

Neil F. Abernethy

Judd L. Walson

Program Authorized to Offer Degree:  
Biomedical Informatics and Medical Education

UMI Number: 3609485

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



UMI 3609485

Published by ProQuest LLC (2014). Copyright in the Dissertation held by the Author.

Microform Edition © ProQuest LLC.

All rights reserved. This work is protected against unauthorized copying under Title 17, United States Code



ProQuest LLC.  
789 East Eisenhower Parkway  
P.O. Box 1346  
Ann Arbor, MI 48106 - 1346



University of Washington

**Abstract**

Ontology-Based Data Integration of Open Source Electronic Medical Record and  
Data Capture Systems

Alicia F Guidry

Chair of Supervisory Committee:

Professor James F Brinkley III

Biomedical Informatics and Medical Education

In low-resource settings, the prioritization of clinical care funding is often determined by immediate health priorities. As a result, investment directed towards the development of standards for clinical data representation and exchange are rare and accordingly, data management systems are often redundant. Open-source systems such as OpenMRS and OpenClinica provide an opportunity to leverage available systems to improve standards and increase interoperability. Nevertheless, continuity of care and data sharing between these systems remains a challenge, particularly in populations with changing health needs, and inconsistent access to health resources.

The overarching goal of this project is to enable sharing of data across low cost systems like OpenMRS and OpenClinica using ontologies. The project consists of three aims: 1) describing clinical research and visit data related to the treatment and care of HIV/AIDS patients, 2) developing a prototype data integration system between electronic medical record and electronic data capture systems, and 3) evaluating the utility of the prototype system using simulated and real-world data. In the first aim, I developed a patient identifier and a HIV/AIDS treatment and care ontology to represent the types of data and information created and used by clinicians. This was achieved by gathering data forms used in

HIV/AIDS clinics in low-resource settings. From these forms, the patient identifier and HIV/AIDS variables were extracted and used to create the ontologies. In aim 2, the ontologies from aim 1, along with simulated data, were used to develop a prototype data integration system that improves the ability of developers to implement integration systems that meet the needs of users, based on previously created use cases. In the third aim, I evaluated whether the matching algorithm used in the prototype can correctly identify matching patients, and whether the prototype is generalizable to clinical care and research data collected in a real world setting.

This work contributes two ontologies to the medical and public health fields that are useful in providing standardization of data elements. Additionally, I provide a prototype data integration system that is useful in facilitating access to previously siloed data and helps reduce the burden of integrating future systems.

# Table of Contents

List of Figures .....	vi
List of Tables .....	ix
List of Abbreviations and Acronyms .....	xi
Chapter 1: Introduction .....	1
1.1 Motivation .....	1
1.2 Problem Statement.....	2
1.3 Approach .....	3
1.4 Research Tasks .....	3
1.5 Contributions.....	4
1.6 Overview of Dissertation .....	4
Chapter 2: Background .....	6
2.1. HIV Disease Management.....	6
2.2 Medical Landscape in Kenya.....	10
2.3 Methods of medical data collection .....	12
2.4. Inability to share data.....	15
2.5. Data standards in Kenya.....	15
2.6. Terminologies and Ontologies in Medicine.....	17
2.7. Unique data in this setting.....	19
2.8. Record Linkage .....	20
2.9. Why ontologies in Kenya?.....	21
2.10. Data Integration.....	22
2.11. Ontologies can aid in data integration.....	24
2.12. Ontology-Based Data Integration Approach .....	26

<b>2.13. Conclusions.....</b>	<b>26</b>
<b>Chapter 3: Aim 1 – Describing Clinical Data .....</b>	<b>27</b>
<b>3.1 Data Collection .....</b>	<b>27</b>
<b>3.2 Patient Identifier Ontology.....</b>	<b>31</b>
3.2.1 Data Model Analysis .....	35
3.2.2 Ontology Design .....	35
<b>3.3 HIV Treatment and Care Ontology .....</b>	<b>39</b>
3.3.1 Data Model Analysis.....	41
3.3.2 Ontology Design .....	41
<b>3.4 Conclusions.....</b>	<b>44</b>
<b>Chapter 4: Aim 2 – Data Integration &amp; Exchange .....</b>	<b>45</b>
<b>4.1 Requirements and Scenarios of Use .....</b>	<b>45</b>
<b>4.2 Prototype System Architecture .....</b>	<b>47</b>
4.2.1 User Interface Component.....	48
4.2.2 Processing Component .....	49
4.2.2.1 U.I. to Ontology Mapping Document .....	49
4.2.2.2 OWL Ontology .....	50
4.2.2.3 Ontology to Database Mapping Document .....	50
4.2.2.4 Data Sources .....	51
4.2.2.5 Data Source Text Files .....	51
4.2.3 Patient Matching Component .....	52
<b>4.3 System Functionality .....</b>	<b>53</b>
<b>4.4 Application Deployment .....</b>	<b>65</b>
<b>4.5 Conclusions.....</b>	<b>65</b>
<b>Chapter 5: Simulated and Real-World Data.....</b>	<b>66</b>
<b>5.1 Simulated Data Creation .....</b>	<b>66</b>

5.1.1 Overview .....	66
5.1.2 Demographic Data.....	66
5.1.3 Clinical Visit Data .....	68
5.1.4 Clinical Research Data .....	69
5.1.5 Data Overlap .....	69
5.1.6 Data Import.....	70
<b>5.2 Real-World Data Sources.....</b>	<b>70</b>
5.2.1 Clinical Visit Data .....	70
5.2.2 Clinical Research Data .....	71
5.2.3 Data Overlap .....	72
5.2.4 Data Import.....	72
<b>5.3 Conclusions.....</b>	<b>73</b>
<b>Chapter 6: Patient Matching Algorithm Evaluation .....</b>	<b>74</b>
<b>6.1 Evaluation Metrics.....</b>	<b>74</b>
<b>6.2 Phase 1: How sensitive is the algorithm given complete data? .....</b>	<b>75</b>
6.2.1 Methods .....	75
6.2.2 Simulated Data Results .....	76
6.2.3 Real World Data Results .....	76
<b>6.3 Phase 2: How much does the sensitivity of the algorithm fade as the data becomes more realistic?.....</b>	<b>77</b>
6.3.1 Methods .....	77
6.3.2 Results .....	79
<b>6.4 Conclusions.....</b>	<b>80</b>
<b>Chapter 7: OBDIS Evaluation.....</b>	<b>82</b>
<b>7.1 Methods .....</b>	<b>82</b>

7.1.1 Scenarios Of Use .....	83
7.1.2 Evaluation Metrics .....	84
<b>7.2 Simulated Data Evaluation Results.....</b>	<b>86</b>
<b>7.3 Real World Data Evaluation Results .....</b>	<b>90</b>
<b>7.4 Adaptability and Extensibility Evaluation.....</b>	<b>94</b>
<b>7.5 Discussion.....</b>	<b>96</b>
<b>7.6 Conclusions.....</b>	<b>97</b>
<b>Chapter 8: Conclusions.....</b>	<b>99</b>
<b>8.1. Limitations .....</b>	<b>99</b>
8.1.1 Limitations in Data Collection and Ontology Development.....	99
8.1.2 Limitations in Integration Methods .....	99
8.1.3 Limitations in Evaluation .....	100
<b>8.2. Future Work.....</b>	<b>100</b>
8.2.1 Save Functionality .....	100
8.2.2 User Interface.....	100
8.2.2.1 Cohort Query Flowchart Interface.....	100
8.2.2.2 Patient Matching Results .....	101
8.2.3 Configuration .....	102
8.2.4 Ontology Refinement.....	103
8.2.5 Data Transfer between systems .....	103
8.2.6 Non-database data sources.....	103
<b>8.3. Deployment Recommendations .....</b>	<b>103</b>
8.3.1 Deployment using a single server with network access to multiple remote sites .....	103
8.3.2 Deployment using no server and multiple remote sites.....	103
<b>8.4. Contributions.....</b>	<b>104</b>
<b>8.5 Summary.....</b>	<b>105</b>

<b>References .....</b>	<b>107</b>
<b>Appendix A. NASCOP Patient Comprehensive Care Card – MOH 257; “Blue Card” .....</b>	<b>116</b>
<b>Appendix B. Patient Identifier Ontology Data Tables.....</b>	<b>121</b>
<b>Appendix C. Patient Identifier Ontology .....</b>	<b>123</b>
<b>Appendix D. HIV Opportunistic Infections .....</b>	<b>133</b>
<b>Appendix E. WHO Clinical Staging Tables .....</b>	<b>134</b>
<b>Appendix F. HIV Ontology Data Tables .....</b>	<b>136</b>
<b>Appendix G. HIV Ontology .....</b>	<b>142</b>
<b>Appendix H. System screenshots by scenario .....</b>	<b>155</b>
<b>Appendix I. Installation Instructions .....</b>	<b>162</b>
<b>Appendix J. Manual and Generated SQL Queries by Scenario Simulated Evaluation .....</b>	<b>163</b>
<b>Appendix K. Manual and Generated SQL Queries by Scenario Real World Evaluation.....</b>	<b>168</b>

# List of Figures

---

FIGURE 1 - HIV COUNSELING AND TESTING PROTOCOL.....	8
FIGURE 2 - SUMMARY OF CLINICAL AND LAB FOLLOW UP OF A PATIENT ON ART .....	10
FIGURE 3 - PROVINCE MAPS OF KENYA SHOWING: A) 100 M RESOLUTION POPULATION DENSITY (TATEM ET AL 2007); B) DISTRIBUTION OF PUBLIC HEALTH FACILITIES IN 2003 (N = 3,048)*; AND C) DISTRIBUTION OF PUBLIC HEALTH FACILITIES IN 2008 (N = 4,933)*. 390 HEALTH FACILITIES THAT WERE NOT SPATIALLY POSITIONED 67 SPECIALIST FACILITIES THAT DO NOT PROVIDE SERVICES TO AMBULATORY PATIENTS ARE NOT SHOWN ON THE BOTH HEALTH FACILITY MAPS. CE = CENTRAL PROVINCE; CO = COAST PROVINCE; EA = EASTERN PROVINCE; NE = NORTH EASTERN PROVINCE; NR = NAIROBI PROVINCE; NY = NYANZA PROVINCE; RV = RIFT VALLEY; WE = WESTERN PROVINCE. *HEALTH FACILITIES THAT FALL WITHIN UNPOPULATED AREAS SUCH PARKS AND GAME RESERVES SERVE STAFF WORKING IN THESE ESTABLISHMENTS AND/OR COMMUNITIES AROUND THE PROTECTED AREAS.....	11
FIGURE 4 - PAIRWISE MAPPING VS. ONTOLOGY MAPPING PER CONCEPT .....	26
FIGURE 5 - PROTEGE 4.1 SCREENSHOT.....	34
FIGURE 6 - PATIENT IDENTIFIER ONTOLOGY - CONCEPTUAL THING CLASS HIERARCHY .....	36
FIGURE 7 - PATIENT IDENTIFIER ONTOLOGY - PHYSICAL THING CLASS HIERARCHY.....	37
FIGURE 8 - VISUAL DIAGRAM OF JURISDICTION AND ADDRESS CLASSES FROM THE PATIENT IDENTIFIER ONTOLOGY.....	39
FIGURE 9 - HIV ONTOLOGY CLASS HIERARCHY .....	42
FIGURE 10 - VISUAL DIAGRAM OF DISEASE, TUBERCULOSIS, DISEASE EPISODE, LABORATORY TEST, HIV AND WHO CLINICAL STAGE CLASSES FROM HIV ONTOLOGY .....	43
FIGURE 11 - SYSTEM ARCHITECTURE BY COMPONENT .....	48
FIGURE 12 - U.I. TO ONTOLOGY MAPPING DOCUMENT; PATIENT ID QUERY SCREEN .....	49
FIGURE 13 - DATABASE TO ONTOLOGY MAPPING DOCUMENT; OPENMRS PERSON TABLE .....	50
FIGURE 14 - PATIENT MATCHING ALGORITHM RESULT; INITIAL ROW ADDED FOR CLARITY.....	53
FIGURE 15 - INDIVIDUAL PATIENT IDENTIFICATION QUERY SCREEN .....	54
FIGURE 16 - COHORT QUERY SCREEN .....	55

FIGURE 17 - SCENARIO-BASED QUERIES MENU SCREEN.....	56
FIGURE 18 - SYSTEM ARCHITECTURE WORKFLOW .....	57
FIGURE 19 - U.I. TO ONTOLOGY MAPPING XQUERY RESULT. U.I. FIELD NAME AND ONTOLOGY CONCEPT SEPARATED BY A "-". .....	57
FIGURE 20 - PTID ONTOLOGY SPARQL QUERY RESULT EXCERPT.....	58
FIGURE 21 – U.I. TO DB MAPPING DOCUMENT QUERY RESULT EXCERPT .....	59
FIGURE 22 - PATIENT DATA QUERY RESULTS SCREEN.....	60
FIGURE 23 - MATCHING COMPARISON RESULTS SCREEN .....	61
FIGURE 24 - PATIENT ENCOUNTER HISTORY SCREEN .....	63
FIGURE 25 - SPECIFIC PATIENT ENCOUNTER SCREEN.....	63
FIGURE 26 - PATIENT VITAL SIGNS AND LAB RESULTS HISTORY .....	64
FIGURE 27 - SIMULATED DATA: CLINICAL RESEARCH VERSUS CLINICAL VISIT VENN DIAGRAM .....	69
FIGURE 28 - REAL WORLD DATA: CLINICAL RESEARCH VERSUS CLINICAL VISIT VENN DIAGRAM .....	73
FIGURE 29 - PHASE 2: AVERAGE MATCH PROBABILITY BY SETTING NUMBER. SETTING NUMBER IS SHOWN ON THE X-AXIS, WHICH THE Y-AXIS SHOWS THE AVERAGE PROBABILITY. ....	80
FIGURE 30 - COHORT QUERY RESULTS SCREENSHOT .....	101
FIGURE 31 - SCENARIO 1, QUERY 2 FIND ALL FUTURE VISIT DATES FOR PATIENT X. QUERY ENTRY FORM SCREENSHOT.....	155
FIGURE 32 - SCENARIO 1, QUERY 2 PATIENT QUERY RESULTS SCREENSHOT .....	155
FIGURE 33 - SCENARIO 1, QUERY 3: IDENTIFY RECENT LABORATORY DATA FOR PATIENT X, QUERY ENTRY FORM SCREENSHOT.....	156
FIGURE 34 - SCENARIO 1, QUERY 3 PATIENT QUERY RESULTS SCREENSHOT .....	157
FIGURE 35 - SCENARIO 2 COHORT QUERY ENTRY FORM SCREENSHOT .....	158
FIGURE 36 - SCENARIO 2 COHORT QUERY RESULTS SCREENSHOT .....	158
FIGURE 37 - SCENARIO 3, QUERY 1 FIND ALL PATIENTS OF PHYSICIAN Y WHO ARE ON DRUG(S) A AND HAVE/HAVE NOT HAD A CLINICAL ENCOUNTER IN THE PAST X MONTHS. QUERY ENTRY FORM SCREENSHOT. ....	159
FIGURE 38 - SCENARIO 3, QUERY 1 PHYSICIAN COHORT RESULTS SCREENSHOT.....	160
FIGURE 39 - SCENARIO 3, QUERY 2 CLINIC QUERY RESULTS SCREENSHOT .....	161

FIGURE 40 - SCENARIO 3, QUERY 2 CLINIC QUERY RESULTS SCREENSHOT ..... 161

# List of Tables

---

TABLE 1 - USA VS. KENYA IDENTIFYING INFORMATION DIFFERENCES .....	20
TABLE 2 - DATA SOURCES, THEIR TYPE AND PRIMARY USER.....	30
TABLE 3 – PATIENT IDENTIFIER DATA MODEL SURVEY – ABBREVIATED .....	32
TABLE 4 - PATIENT IDENTIFIER ONTOLOGY DATA FIELD CATEGORIES AND EXAMPLES.....	33
TABLE 5 - HIV ONTOLOGY DATA FIELD CATEGORIES AND EXAMPLES.....	40
TABLE 6 - SCENARIOS OF USE (* SCENARIOS THAT ARE NOT IMPLEMENTED IN THE PROTOTYPE SYSTEM.).....	47
TABLE 7 - SIMULATED DATA EXTERNAL DATASETS.....	67
TABLE 8 - PHASE 1 SIMULATED DATA: PERCENTAGE OF TIMES PATIENT (%) WAS FOUND IN THE LIST BY QUESTION .....	76
TABLE 9 - PHASE 1 REAL WORLD DATA: PERCENTAGE OF TIMES PATIENT (%) WAS FOUND IN THE LIST BY QUESTION.....	77
TABLE 10 - PHASE 2: PATIENT-MATCHING ALGORITHM EVALUATION SETTINGS AND OCCURRENCES .....	78
TABLE 11 - PHASE 2: PERCENTAGE OF TIMES PATIENT (%) WAS FOUND IN THE LIST BY SETTING NUMBER AND QUESTION.....	79
TABLE 12 - PHASE 2: AVERAGE PROBABILITY AND POSITION OF CORRECT MATCH BY SETTING NUMBER.....	79
TABLE 13 - SCENARIOS OF USE BY QUERY NUMBER (* QUERY 7 WAS NOT IMPLEMENTED IN THE PROTOTYPE SYSTEM.).....	84
TABLE 14 - METRICS USED TO EVALUATE PROTOTYPE SYSTEM.....	84
TABLE 15 - LIST OF QUERIES, QUESTIONS AND ASSOCIATED METRICS.....	85
TABLE 16 - QUERY 1: AVERAGE TIME REQUIRED TO RETURN RESULTS (IN SECONDS) .....	86
TABLE 17 - QUERY 1: TOTAL NUMBER OF PATIENT ENCOUNTERS .....	86
TABLE 18 - QUERY 2: AVERAGE TIME REQUIRED TO RETURN RESULTS (IN SECONDS) .....	87
TABLE 19 - QUERY 2: TOTAL NUMBER OF SCHEDULED VISITS AFTER JANUARY 1, 2011 .....	87
TABLE 20 - QUERY 3: AVERAGE TIME REQUIRED TO RETURN RESULTS (IN SECONDS) .....	87
TABLE 21 - QUERY 3: TOTAL NUMBER OF LABORATORY DATA POINTS.....	87
TABLE 22 – QUERY 4: AVERAGE TIME REQUIRED TO RETURN RESULTS (IN SECONDS) .....	88

TABLE 23 - QUERY 4: TOTAL NUMBER OF PATIENTS RETURNED.....	88
TABLE 24 - QUERY 5: AVERAGE TIME REQUIRED TO RETURN RESULTS (IN SECONDS) .....	89
TABLE 25 - QUERY 5: TOTAL NUMBER OF PATIENTS RETURNED.....	89
TABLE 26 - QUERY 6: AVERAGE TIME REQUIRED TO RETURN RESULTS (IN SECONDS) .....	89
TABLE 27 - QUERY 6: TOTAL NUMBER OF PATIENTS RETURNED.....	89
TABLE 28 - SIMULATED DATA PRECISION, RECALL AND F-MEASURE BY QUERY .....	90
TABLE 29 – QUERY 1: AVERAGE TIME REQUIRED TO RETURN RESULTS (IN SECONDS) .....	91
TABLE 30 - QUERY 1: TOTAL NUMBER OF PATIENT ENCOUNTERS .....	91
TABLE 31 - QUERY 2: AVERAGE TIME REQUIRED TO RETURN RESULTS (IN SECONDS) .....	91
TABLE 32 - QUERY 2: TOTAL NUMBER OF SCHEDULED VISITS AFTER JANUARY 11, 2010 .....	91
TABLE 33 - QUERY 3: AVERAGE TIME REQUIRED TO RETURN RESULTS (IN SECONDS) .....	92
TABLE 34 - QUERY 3: TOTAL NUMBER OF LABORATORY DATA POINTS .....	92
TABLE 35 – QUERY 4: AVERAGE TIME REQUIRED TO RETURN RESULTS (IN SECONDS) .....	92
TABLE 36 - QUERY 4: TOTAL NUMBER OF PATIENTS RETURNED.....	92
TABLE 37 - QUERY 5: AVERAGE TIME REQUIRED TO RETURN RESULTS (IN SECONDS) .....	93
TABLE 38 - QUERY 5: TOTAL NUMBER OF PATIENTS RETURNED.....	93
TABLE 39 - QUERY 6: AVERAGE TIME REQUIRED TO RETURN RESULTS (IN SECONDS) .....	94
TABLE 40 - QUERY 6: TOTAL NUMBER OF PATIENTS RETURNED.....	94
TABLE 41 - REAL WORLD DATA PRECISION, RECALL AND F-MEASURE BY SCENARIO AND QUERY.....	94

---

# List of Abbreviations and Acronyms

---

AIDS	Acquired Immunodeficiency Syndrome
AMPATH	Academic Model Providing Access to Healthcare
AMRS	AMPATH Medical Record System
API	Application Programming Interface
ART	Anti-retroviral Therapy
ARV	Anti-retroviral
BDIS	Biomediator Data Integration System
CD4	Cluster of Differentiation 4
CDC	Centers for Disease Control and Prevention
CDC-CSTE	Centers for Disease Control and Prevention – Council of State and Territorial Epidemiologists
CDISC-ODM	Clinical Data Interchange Standards Consortium-Operational Data Model
CIS	Clinical Information System
CRF	Case Report Form
CSV	Comma Separated Value
CTMS	Clinical Trials Management System
DB	Database
DOB	Date of Birth
EDC	Electronic Data Capture
EM	Expectation Maximization
EMR	Electronic Medical Records
FACES	Family AIDS Care and Education Services
FMA	Foundational Model of Anatomy
HIV	Human Immunodeficiency Virus
HIVO	Human Immunodeficiency Virus Ontology
HL7	Health Level Seven

HTML	Hypertext Markup Language
I2b2	Informatics for Integrating Biology and the Bedside
ICD-10	International Statistical Classification of Diseases and Related Health Problems
ID	Identification Number
IDO	Infectious Disease Ontology
IML	Immediate Language
ISO	International Standards Organization
IT	Information Technology
I-TECH	International Training and Education Center for Health
KEBS	Kenya Bureau of Standards
KEMRI	Kenya Medical Research Institute
LOINC	Logical Observational Identifiers Names and Codes
M&E	Monitoring and Evaluation
MMRS	Mosoriot Medical Records System
MOH	Ministry of Health
MVP/CIEL	Millennium Villages Project/Columbia International eHealth Laboratory
LCS	Longest Common Substring
MOH	Ministry of Health
MPI	Master Patient Index
NASCOP	National AIDS and STD Control Programme
NCI	National Cancer Institute
NGO	Non-governmental Organization
OBDIS	Ontology-Based Data Integration System
ODBC	Open Database Connectivity
OCRe	Ontology of Clinical Research
OI	Opportunistic Infection
OS	Operating System
OWL	Web Ontology Language

PEPFAR	U.S. President's Emergency Plan for AIDS Relief
PHE/THE	Empiric Therapy of Helminth Co-Infection to Reduce HIV-1 Disease Progression
PHP	Hypertext Preprocessor
PIDO	Patient Identifier Ontology
PQL	Program Query Language
QES	Query Execution Service
QI	Query Integrator
RAM	Random Access Memory
RAND	Research and Development
REST	Representational State Transfer
RDF	Resource Description Framework
SERO	Southeast Asia Regional Offices
SIG	Structural Informatics Group
SKB	Source Knowledge Base
SNOMED-CT	Systemized Nomenclature of Medicine – Clinical Terms
SPARQL	SPARQL Protocol and RDF Query Language
SSN	Social Security Number
SQL	Structured Query Language
THE	Empiric Therapy of Helminth Co-Infection to Reduce HIV-1 Disease Progression
UI	User Interface
UMLS	Unified Medical Language System
UNAIDS	United Nations Programme on HIV/AIDS
UPI	Unique Patient Identifier
URL	Uniform Resource Locator
USAID	United States Agency for International Development
VCT	Voluntary Counseling and Testing
vSPARQL	View SPARQL Protocol and RDF Query Language
WHO	World Health Organization

XML	eXtensible Markup Language
XSD	XML Schema Definition
XSL	eXtensible Stylesheet Language

## Acknowledgements

To my parents Eddie and Shelby Guidry, and my sister Michelle, I appreciate your endless encouragement, patience, and support. Thank you for listening to me talk through my ontologies and Java problems even though I am more than sure you thought I was speaking a foreign language.

To my committee, thank you for your mentorship. It took a lot longer than we expected, but I am finally finished. I have learned more than I could have ever imagined and will use this knowledge to further my research career.

To Dr. Isaiah Warner, Monica, Lisa and Karin, thank you for getting me to this point and encouraging me to pursue a PhD. To Dr. Sonja Wiley-Patton thank you for introducing me to Health Informatics and applied Computer Science.

To Dr. Judd Walson, Ben Piper and the Kenya Medical Research Institute –UW staff thank you for giving me an opportunity to visit and really see how healthcare is managed in Kenya. My time in Kenya was instrumental in changing the course of my research. To Edwin, Linda and Rose, thank you for your friendship, showing me your world and making yourself available to answer my endless list of questions.

To my classmates Drs. Wynona Black, Daniel Capurro, Michal Galdzicki, Rebecca Hills, Casey Overby, Blaine Reeder, and Clara Wilkins, and Melissa Clarkson thank you for your support and feedback. I would like to extend a special thanks to Sandy Turner and the BHI staff for their help in navigating the dissertation process.

To my family, friends, and the Church of the Burning Fire, thank you for your prayers and encouragement. I could not have done this without you.

I would also like to acknowledge the funding that made this work possible. This project was funded in part by the UW Biomedical and Health Informatics training grant (NIH T15 LM07442) and by (Judd's Grant). Finally, I would like to thank Dr. James Berkley from the KEMRI/Wellcome Trust Research Programme for helping obtain access to the EMR data used in this study.

## **Dedication**

To my parents Eddie and Shelby Guidry, and my grandmothers Leola Guidry and Mary Martin

# Chapter 1: Introduction

## 1.1 Motivation

The healthcare landscape in many developing countries is often comprised of multiple agencies supporting the delivery of medical care. These avenues include, but are not limited to: private providers; local ministries of health (MOH); and non-governmental organization (NGO)-based clinics. Moreover, these clinics can be financially supported by multiple internal or external agencies, such as the United States Agency for International Development (USAID), President's Emergency Plan for AIDS Relief (PEPFAR), Centers for Disease Control and Prevention (CDC), and the World Health Organization (WHO). Often, individual funders focus on specific diseases or conditions, particular populations, or fixed regions. They also draw from distinct funding sources and have distinct monitoring and evaluation (M&E) requirements.

In much of sub-Saharan Africa, there has been a massive response to the emergence of the Human Immunodeficiency Virus and Acquired Immunodeficiency Syndrome (HIV/AIDS) with extensive funding from the Kenyan MOH and many outside sources. Kenya spent \$1,629 million on health expenditures in 2009/2010 and a quarter of this expenditure (\$397.5 million), was spent on HIV and AIDS. More than half (51%) of this funding was provided by external donors (1). These funds have been directed towards public education to prevent new transmission and encourage testing, treatment and care for those affected by the virus and for research. While the number of new cases has been on a "slow and steady decline" since 2008, the virus continues to be a significant burden on a healthcare system that is already strained (2).

As a result of this massive scale up of health care delivery and the multiple distinct funding sources, many delivery sites collect their own data, and if the site is computerized, has its own associated information system. This approach, while perhaps effective for any single provider system, does not benefit the healthcare system as a whole. In addition, such systems may not be able to track patients when a patient moves or requires overlapping care from multiple systems. These constraints often cause healthcare information to become isolated within a single provider's system, resulting in unnecessary

redundancy of effort and resources, duplicate data entry, fragmented health records, and poor allocation of time for both patients and clinicians (3,4).

In an effort to mitigate the effects of the currently fragmented system, Kenya is working to implement a country-wide electronic medical record (EMR) system (4). While such a country-wide information system alone cannot improve access to care, it may help to mitigate the impact of some of these issues, by providing data standards and system interoperability. The development and implementation of this EMR is currently underway; however, it has not been fully realized and may not be for some time. As of January 2013, ten health facilities are currently using the system and fifteen additional sites were expected to begin using the system by the end of March 2013. The current plan calls for a deployment of 300 sites over the next two years.<sup>1</sup> However, to be useful, this implementation will require the extraction, transformation and loading of existing data from legacy systems and data from existing paper records will also need be entered into the EMR. While these processes can be both timely and costly, the lack of data integration poses real concerns for optimal patient care in Kenya.

## 1.2 Problem Statement

For populations with changing health needs and inconsistent access to health resources, continuity of care and data sharing between systems remain a challenge. In addition to health-related problems, data sharing also presents an informatics problem, mapping data fields between multiple systems. Each variable in a data source, health information system, must be mapped pairwise to the corresponding variable(s) in every other data source. This process increases the amount of time required to build data sharing functionality. Furthermore, limited funding requires that potential solutions be low-cost and appropriate to the needs of users. The importance of funding is particularly relevant in HIV care and prevention because of the number of people affected by the virus as well as the complex protocols required to treat the virus.

---

<sup>1</sup> <https://wiki.ampath.or.ke/display/forms/OpenMRS+Kenya>

### 1.3 Approach

This research will seek to identify the optimal method for developing an ontology-based data integration system (OBDIS) for use in the treatment and care of HIV/AIDS patients. These ontologies potentially provide a more efficient alternative to pairwise mapping between each source's variables. Instead, each source's variables will be mapped to the ontologies, enabling data mappings where these had previously been infeasible between systems and reducing the time required to build data sharing functionality. In addition, the ontologies and mappings will reduce or eliminate the need for custom programming, thereby decreasing the amount of time it takes to access a new source. This approach will leverage currently available systems, open source EMR and electronic data capture (EDC) systems, in order to improve standards and increase interoperability.

### 1.4 Research Tasks

This research will answer the following questions:

- Can clinical trial management (CTMS) and EMR systems be faithfully represented by formal knowledge representation (ontology)?
- Can this knowledge representation be used to facilitate data integration and exchange between open source EMR and EDC systems, specifically the ability to query between systems?
- Can an ontology-based approach help to reduce the amount of time required to develop integration systems that meet the needs of users based on previously created use cases and requirements?
- To what extent can the resulting functionality meet the information needs of researchers, clinicians and all tiers of health care support who use one or more systems (e.g., providing accurate patient information)?

These questions will be answered through the completion of three research aims: 1) describing clinical research and visit data related to the treatment and care of HIV/AIDS patients, 2) developing a prototype data integration system between EMR and EDC systems, and 3) evaluating the utility of the prototype system. In the first aim, I developed two ontologies, patient identifier, and HIV/AIDS - to

represent the types of data and information created and used in the treatment and care of HIV/AIDS patients. The second aim uses the ontologies from Aim 1 to implement an OBDIS that improves the ability of developers to implement integration systems that meet the needs of users based on previously created use cases and requirements. In Aim 3, I conducted two evaluations using simulated and real world data from Kenya. First, I evaluated the patient matching algorithm, used in the prototype system, to determine the algorithm's performance when matching patients, without a common identification number (ID), across systems in various real world conditions. Finally, I evaluated whether the prototype system accurately returns data and provides functionality that meets the requirements for an OBDIS in low-resource settings and those of the scenarios of use we have created. The generalizability of the approach is also evaluated in Aim 3.

## **1.5 Contributions**

This project uses disease specific, HIV/AIDS and patient identifier ontologies to facilitate data integration and exchange between two open source systems popular in low-resource areas and globally, and shows how the use of ontologies eliminates the need to pairwise map between information systems. It also helps to provide better coordination of care based on the data generated by both clinical research and clinical care. Additionally, adoption of ontologies for use in developing information systems can encourage implementers to conform to the data standards set forth by the ontology. Finally, over time this system can help to reduce duplicate and unnecessary medical tests by providing clinicians and researchers with access to data and information, which would have not been easily accessible previously.

## **1.6 Overview of Dissertation**

The organization of this dissertation is as follows. Chapter 2 begins with a description of HIV, the medical landscape in Kenya, and the usage of open source software in medicine. Next, I describe record linkage and the problems associated with its use in low-resource settings, followed by a description of terminologies and ontologies, in the context of this work. Chapter 2 closes with a discussion of data integration and associated methods.

The basis for this research, the ontologies, HIV/AIDS and patient identifier, will be described in Chapter 3. I will discuss the data collection process, followed by a description of how the ontologies were created and a description of their content. Completed data collection tables and full visual representations are included as appendices for both ontologies.

In Chapter 4, I will review the requirements and scenarios of use for an OBDIS in Kenya. Finally, I provide a detailed description of the prototype version of the data integration system that includes screenshots of the system.

The final phase of this dissertation is the evaluation, which I will detail in Chapters 5, 6, and 7. Chapter 5 will describe the simulated and real world data used in these evaluations as well as how they were created and obtained. Chapters 6 and 7 will detail how the approach used in Chapter 4 was evaluated using simulated and real world data from Kenya. Specifically, Chapter 6 includes a description of the two-phase evaluation conducted using the patient matching algorithm. Chapter 7 focuses on the evaluation of the prototype system as a whole. Two appendices that list the manual and SQL general queries used in this evaluation supplement this chapter.

In the final chapter, Chapter 8, I discuss the limitations of this research and areas of future work, including real world implementation recommendations. Finally, I provide a summary of my results and discuss the project's contributions to informatics and medical care in low-resource settings.

## Chapter 2: Background

This chapter will help to place the contributions of this work, the ontologies and the prototype system, in the context of the settings in which I believe this work to be beneficial. I will briefly describe HIV epidemiology and the healthcare system in Kenya. Next, I discuss the use of open source software in medicine, specifically in low-resource settings. I will also review the current work related to terminologies and ontologies, record linkage, and data integration. Finally, I will conclude with a discussion of the challenges associated with data integration in Kenya and my approach to solving this problem.

### 2.1. HIV Disease Management

HIV is a “retrovirus that infects cells of the immune system, destroying or impairing their function” (5). “HIV is transmitted through unprotected sexual intercourse, transfusion of contaminated blood, sharing of contaminated needles, and between a mother and her infant during pregnancy, childbirth and breastfeeding” (6). There are two major types of HIV virus infections, HIV-1 and HIV-2. “HIV-1 is the cause of much of the global HIV pandemic and is much more infective than HIV-2” (7). HIV-2 progresses more slowly than HIV-1 and is found primarily in West Africa. HIV-1 infection, in the absence of effective treatment, usually progresses to AIDS (5). In the 2010 AIDS epidemic update, the Joint United Nations Programme on HIV/AIDS (UNAIDS), estimated that there are 33.3 million people living with HIV in the world, 15 million of which are from low- and middle-income countries (8). It is important to note that these 33.3 million individuals are at risk for developing AIDS as well as one or more of the many opportunistic infections (OI) associated with the disease, further increasing the burden of disease and complicating care of infected individuals. OIs are infections that take advantage of the compromised immune system caused by HIV. A list of OIs associated with HIV can be found in Appendix D (9). Unless otherwise stated, all future references to HIV will be related to the HIV-1 subtype.

In 2009, the WHO estimated the population of Kenya to be 40,513,000 (10). Of those, the Kenya Demographic and Health Survey by the National AIDS Control Council and National AIDS and STD Control Programme (NAS COP), tasked by the Kenya Ministry of Health to lead the government's

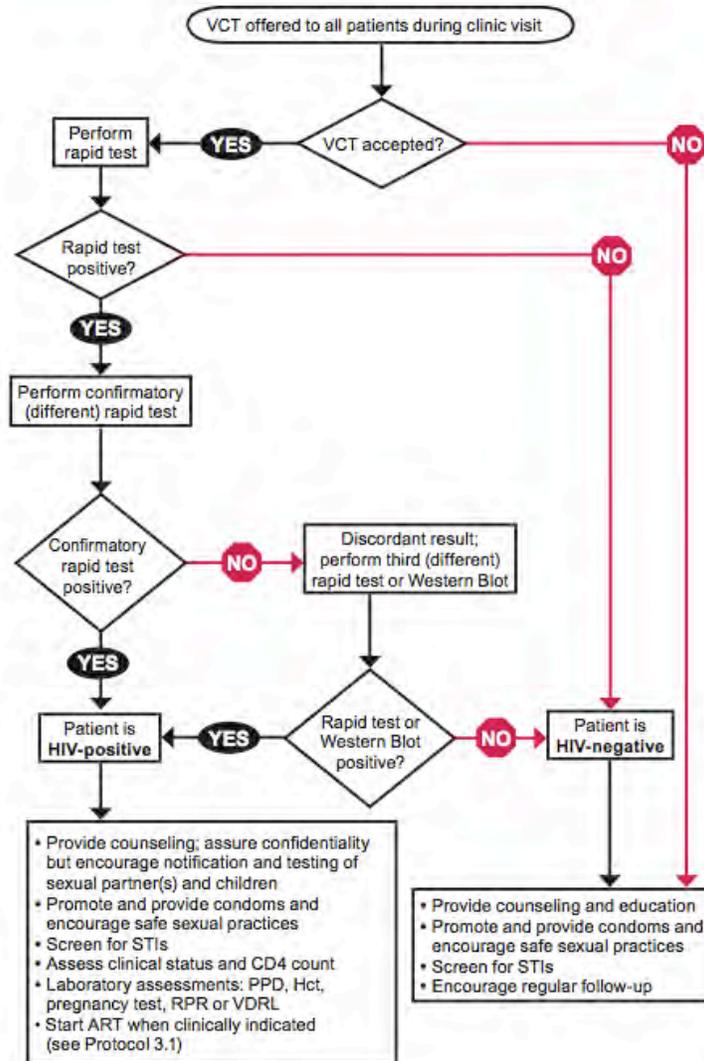
response to the HIV/AIDS Pandemic, estimated 6.3 percent of persons aged 15-49 have HIV (11). This means that 1.3 million to 1.6 million people are living with HIV in Kenya (11). While the number of new HIV infections is declining in Kenya, it is anticipated that new infections among people over the age of fifteen will still be 81,972 in 2013, down from 91,000 in 2011 (2).

Patients are often diagnosed with HIV during clinical visits or at VCT (voluntary counseling and testing) centers. Following an initial test to determine if a patient has antibodies to HIV, a confirmatory test is run. If the result of the first test is negative, the patient is considered to be HIV-negative and encouraged to return for follow-up testing. If the first test is positive, a second rapid test of a different brand is conducted. If both tests one and two are positive, the patient is diagnosed as HIV-positive and protocols for treatment are initiated. If the result of the first test is positive and the second is negative, the result is considered discordant and a third rapid test or Western Blot is conducted. If the third test is negative, the patient is considered HIV-negative and encouraged to return for follow-up testing. If the test is positive, treatment protocols are initiated. These treatment protocols are as follows:

1. Counseling;
2. Promoting and encouraging safe sex practices;
3. Screening for sexually transmitted infections (STIs);
4. Determining clinical status and CD4 count (marker of immune status and HIV disease progression);
5. Conducting laboratory assessments including liver/kidney function and pregnancy tests in female patients; and
6. Starting ART (antiretroviral therapy) when indicated by national guidelines.

A visual flowchart of this process can be found in Figure 1 below, reproduced from the PIH (Partners In Health) Guide to the Community-Based Treatment of HIV in Resource Settings (12).

### Protocol 2.1 Provider-Initiated HIV Counseling and Testing for Adults and Adolescents



**Figure 1 - HIV Counseling and Testing Protocol**

Upon infection of the HIV virus, patients often experience flu-like symptoms; however, it can take between two and eight weeks for the antibodies to be detectable in diagnostic tests. This is considered the “window period” (7). Next, the body’s immune response causes the patient’s viral load to decline and stabilize within the first six to twelve months after infection. Following infection, there is usually an asymptomatic period of six to ten years, during which the patient may be unaware of his/her infection status. After this time period, the patient begins to show symptoms of HIV and its associated OIs.

The WHO has defined four stages based on clinical parameters to reflect the severity and prognosis of the disease (7). In addition to allowing “harmonization in clinical case and surveillance definitions,” staging also facilitates clinical decision making related to ART (7). Clinical staging is based on the patient’s physical status as well as any OI related symptoms or diagnoses (7). The current clinical staging tables for adults and adolescents, and pediatric patients can be found in Appendix E (7).

In Kenya, if it has been determined that the patient’s WHO Clinical Stage is 1 or 2, ART is initiated once a CD4 cell count less than or equal to 350 cells/mm<sup>3</sup> has been detected. Alternatively, if the patient’s WHO Clinical Stage is 3 or 4, ART is initiated at any CD4 cell count. ART is a treatment protocol made up of three classes of antiretroviral (ARV) drugs: Protease inhibitors; Non-nucleoside reverse transcriptase inhibitors; and Nucleoside and nucleotide reverse transcriptase inhibitors (7). First-line treatments are given to patients upon ART initialization. In addition to the standard ART regimen for those infected with HIV, pregnant mothers and those in discordant relationships are also given ART, although a different regimen. Prevention of mother-to-child transmission (PMTCT) is the intervention that provides ART as well as counseling for expectant mothers. The ART regimen given to expectant mothers is different depending on the mother’s exposure to antiretroviral drugs (7). Additionally, the results of a clinical study in Kenya and Uganda, pre-exposure prophylaxis (PrEP), has proven to be effective and may be used for prevention in discordant couples, where one partner has HIV and one does not (13,14).

Patients are expected to maintain a strict adherence >95% as determined by “taking the correct dose of drugs at the correct times while observing any dietary or fluid restrictions” (7). Non-adherence can result in accelerated disease progression, transmission of infection due to increased viral load, and onset of OIs. Side effects, often associated with ARV drugs, disease progression based on WHO Clinical Stages, the onset of OIs, pregnancy or treatment failure, are cause for drug substitutions within the patient’s current first-line regimen or transfer to second-line treatment.

Clinically, patients are monitored at baseline, ART, initialization, two weeks after starting ART and monthly thereafter. Figure 2 below, obtained from the Guidelines for Antiretroviral Therapy in Kenya 4th edition shows the follow up protocol for ART patients (7). Changes in treatment and OIs increase the complexity of treatment protocols and require the patient and their physician(s) deal with an increasing amount of data in an effort to manage the disease and provide comprehensive patient care.

	Week		Month									
	0	2	1	2	3	4	5	6	9	12	Stable	
Appointment*												
Clinical evaluation, Wt, Ht <sup>1</sup> , ADRs	+	+	+	+	+	+	+	+	+	+	+	Every visit
TB screening	+	+	+	+	+	+	+	+	+	+	+	Every visit
Adherence check	+	+	+	+	+	+	+	+	+	+	+	Every visit
Hb	+		+ <sup>2</sup>		+ <sup>2</sup>	Symptom directed						
ALT	+		+ <sup>3,4</sup>		+ <sup>3,4</sup>	Symptom directed						
Creatinine <sup>5</sup>	+	Symptom directed										
Pregnancy test <sup>6</sup> (PT)	+	If indicated										
Urinalysis	+	Symptom directed										
Fasting lipid profile & glucose <sup>7</sup>	+	Annually for patients on PIs										
CD4 count	+							+		+		Every 6 months
Viral load <sup>8</sup>		Targeted										

<sup>1</sup> Weight and height should be measured in children regularly and in adults for BMI calculation at initial assessment

<sup>2</sup> Schedule when AZT is used

<sup>3</sup> Schedule when NVP is used

<sup>4</sup> Schedule in pregnant women: ALT should be done at baseline, 2, 4 weeks then monthly until the woman delivers; especially important in women with CD4 >250 cells/mm<sup>3</sup> at ART initiation on NVP-based regimen

<sup>5</sup> All pts should have creatinine measured if available. NRTI doses may need adjustment if renal function (RF) abnormal. TDF should be avoided if RF abnormal (See Table 20.11-20.14).

<sup>6</sup> PT should be done at baseline if EFV is to be used; thereafter PRN

<sup>7</sup> Schedule if PIs used

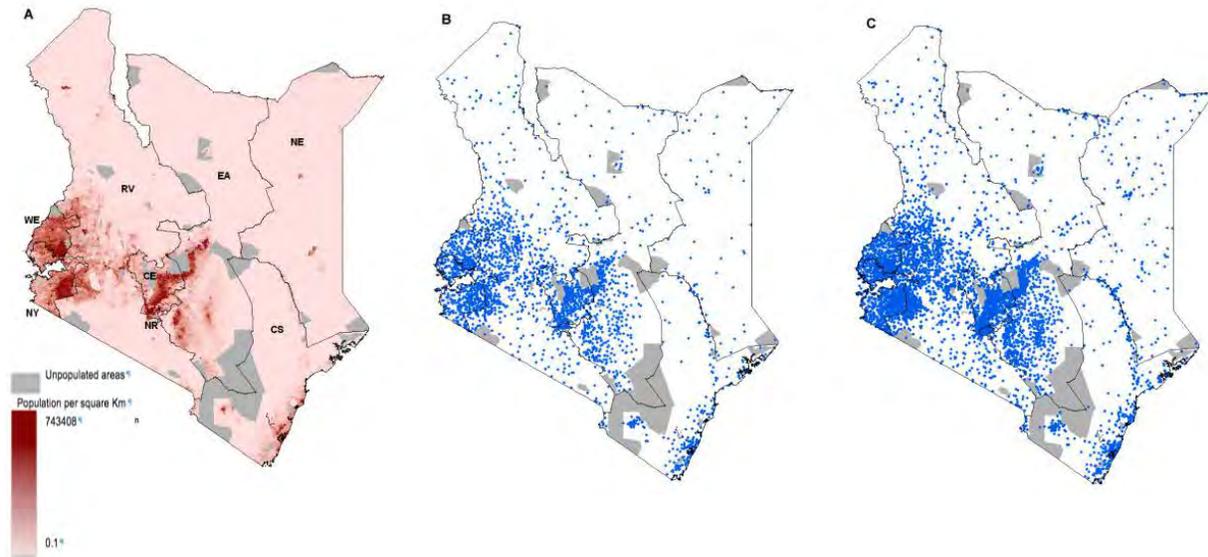
<sup>8</sup> Currently viral load is indicated for suspected treatment failure cases and before substituting d4T in cases of toxicity where d4T has been used for more than 6 months.

\* appointment visits should be based on patients clinical status.

**Figure 2 - Summary of clinical and lab follow up of a patient on ART**

## 2.2 Medical Landscape in Kenya

In order to help these patients manage their disease, HIV clinics, public and private, are available throughout the country. Figure 3 below includes three maps that provide evidence to the number of health facilities in Kenya in 2008 (15). Figure 3A shows the population density of Kenya at 100 m. Figure 3B and 3C show the number of health facilities in Kenya during 2003 (n = 3,048) and 2008 (n = 4,933), respectively.



**Figure 3 - Province maps of Kenya showing: A) 100 m resolution population density (Tatem et al 2007); B) distribution of public health facilities in 2003 (n = 3,048)\*; and C) distribution of public health facilities in 2008 (n = 4,933)\*. 390 health facilities that were not spatially positioned 67 specialist facilities that do not provide services to ambulatory patients are not shown on the both health facility maps. CE = Central province; CO = Coast province; EA = Eastern province; NE = North Eastern province; NR = Nairobi province; NY = Nyanza province; RV = Rift Valley; WE = Western province. \*Health facilities that fall within unpopulated areas such as parks and game reserves serve staff working in these establishments and/or communities around the protected areas.**

The healthcare system in Kenya is tiered, with each tier providing preventative, curative or rehabilitative services, or a combination of the three. The tiers are as follows (lower to higher): Community: Village/households/families/individuals; Dispensaries/clinics; Health centres, maternities, nursing homes; Primary hospitals; Secondary hospitals; and Tertiary hospitals (16). Following the recent elections in Kenya during 2013, the country is moving to reorganize the health care system under “devolution” with decentralization of health care facilities into Counties (17).

In addition to the various types and numbers of clinics, centres and hospitals, research programs also have an opportunity to provide clinical care to the patient. Often, both recruitment and daily research operations are done through health care facilities, but the data generated from these activities are not always made available for use outside of the research program itself. A search using the term “Kenya” in

the WHO International Clinical Trials Registry returns 1,421 registered trials (18). If one considers the number of registered trials and the possibility that there could be hundreds, if not thousands, of unregistered trials, collecting medical data from patients and the 5,712 health institutions in Kenya (as of 2008), the potential for duplicated effort increases (19). More importantly, the number of health institutions does not provide an accurate estimate as to the number of health information systems. Within a facility each clinic (HIV, Tuberculosis (TB), Pediatric, etc.) may have a separate information system. (E.g. the HIV Clinic within a district hospital could have its own system that is completely separate from that of the hospital itself). Given the overlap in geography and population served, the fragmentation of medical records supports the need for data integration.

### **2.3 Methods of medical data collection**

Each clinic, hospital, and research program has its own means of recordkeeping, whether paper, electronic, or a hybrid of the two. There are examples, such as the Academic Model Providing Access to Healthcare (AMPATH), Family AIDS Care and Education Services (FACES) and Millennium Villages Project, where not only are the information systems the same, but the data are also shared between installations. AMPATH's (funded by PEPFAR through USAID) initial purpose was to provide health care to HIV/AIDS patients in Kenya; however, the program has expanded to include, but is not limited to TB, malaria, chronic diseases, as well as maternal and child health (20). The program originally used the AMPATH medical record system (AMRS), but has since switched to the OpenMRS system (21). They have "collected more than 100 million discrete clinical observations from 2.8 million AMPATH visits made by 300,000 enrolled patients" (20). Wireless connectivity between sites was expected to be completed by the end of 2011, to aid in real-time access to patient information.

The FACES program provides family-oriented HIV prevention, care and treatment in Kenya. Funded by PEPFAR through the CDC as well as USAID and the Clinton foundation, it is a collaboration between the University of California San Francisco and the Kenya Medical Research Institute (KEMRI), which supports over 130 sites throughout Kenya. The program has implemented the OpenMRS system in nineteen of those sites in Nyanza Province and Nairobi (22).

Finally, the Millennium Villages project has eighty "villages" in ten Sub-Saharan African countries

(23). The Villages project was started to demonstrate that the Millennium Development Goals could be fully realized. To this end, they do not have a disease focus, but a desire to end poverty in rural Africa by providing access to technologies to enhance “farm productivity, health, education and access to markets.”

(23). They have also implemented the OpenMRS system and are working towards widespread adoption across all “villages” which includes making patient data available across all sites (24).

Despite the gaining popularity of EMRs throughout the country, paper-based methods are still prevalent. The number of EMRs in Kenya will continue to increase as the Kenya Ministry of Health’s initiative to implement a country-wide EMR begins to take shape. In collaboration with International Training and Education Center for Health at the University of Washington and the University of California San Francisco (I-TECH) and OpenMRS, work has begun to develop this EMR. This EMR will be OpenMRS based, and customized to fit the needs of the Kenyan healthcare system.<sup>2</sup> Because this work is not completed, the focus of this dissertation will be on the other EMRs, which can be found in the country. These EMRs are often homegrown, created by the organization itself, or open source. When adopting new information systems, implementers and developers often choose open source solutions because of their ability to provide much needed functionality at little or no cost. Another advantage of open source software is that it allows developers to modify the system to meet their organization’s needs. Finally, popular open source projects have the added benefit of a large developer and implementer community. This community responds to user questions, provides feedback on the project and helps to develop new and existing system modules.

There are many popular open source EMRs including OpenMRS, World VistA, and OpenEHR. OpenMRS, one of the most popular, will be used in this research. It is currently being used on four continents and in over thirty countries (25). OpenMRS was built by the Regenstrief Institute to help manage HIV patient care in low-resource settings, initially Kenya. A major advantage of OpenMRS is that new data forms can be implemented without programming knowledge. The cornerstone of the system is its concept dictionary, which helps to standardize data within a specific implementation. OpenMRS comes with a concept dictionary; however, many implementers have opted to use the Millennium Villages

---

<sup>2</sup> Information about the system’s development process and a link to the most current release can be found at <http://openmrskenya.blogspot.com>.

Project/Columbia International eHealth Laboratory (MVP/CIEL) Concept Dictionary (23). This dictionary was developed for the Millennium Villages Project's OpenMRS installations in addition to concepts related to the data entry forms used in their clinics, maps to the Systemized Nomenclature of Medicine – Clinical Terms (SNOMED-CT), and International Statistical Classification of Diseases and Related Health Problems (ICD-10) terminologies.

Since the initial development of the system in 2004, the OpenMRS implementations throughout the world have provided lessons learned which have proven useful on many occasions. Specifically, the limitations of language translation, reliable electricity, funding, and staffing were among those issues which remain challenges to implementing the system (26–28). Moreover, these studies cited not only the importance of access to data for patient care and reporting purposes, which is also a limitation due to the lack of reliable electricity, but also the need for standardization of data to limit data entry errors.

Alternatively, open source EDC systems, used for clinical trial data collection, are available (e.g. OpenClinica and REDCap) and widely used throughout the world. However, most researchers use spreadsheets and homegrown systems (29,30). While homegrown systems may meet the needs of individual researchers, spreadsheets are limited in their ability to provide complex data queries, data validation or protection against data corruption, and functionality that most databases offer. Managing a study using spreadsheets often requires multiple worksheets and workbooks, which contain thousands of rows and data columns. Spreadsheets do provide useful functionality, such as search and customization. However, as size (e.g. number of rows, columns and worksheets) increases the ease of interacting with and manipulating, the data decreases (30). Moreover, homegrown systems and spreadsheets are not often scalable to meet the needs of its users beyond initial use.

OpenClinica, created by Akaza Research to help clinical researchers collect and manage clinical trial data, was also used in this research. The software allows users to implement data forms without programming knowledge. Though OpenClinica does not require the use of concept dictionaries or medical terminologies, it does implement Clinical Data Interchange Standards Consortium-Operational Data Model (CDISC-ODM). CDISC-ODM provides audit trail capabilities, but most importantly, it provides a way to import and export data in eXtensible Markup Language format (XML). While this format does not require a controlled vocabulary or terminology, it does standardize the data being imported and

exported by providing a way to uniformly structure a clinical trial as well as the variables and data collected.

## **2.4. Inability to share data**

Despite the use of open source and homegrown software, most systems lack data standards, which limit the ability to share data across systems (aside from entering the data by hand). Providing access to data and information not otherwise available, like the data in OpenClinica and OpenMRS, reduces the need for duplicate tests and redundant data, because these systems are often siloed. However, the ability to share data in low-resource settings is limited by the use of paper-based data forms. These forms are often completed by hand, and if an electronic system exists, the data are later entered into the system (31,32).

While paper-based data management is low-cost and allows for quick implementation (minimal training), it is limited in its ability to aid in providing decision support at the point of care. Paper records are also subject to errors, lack of data standards, illegible handwriting, and are hard to search, resulting in patient care errors or lack of information for use in decision-making (33). Finally, paper records can be misplaced or destroyed within the agency or during transport between agencies, if necessary, resulting in incomplete patient data (34).

## **2.5. Data standards in Kenya**

In Kenya, when a patient is diagnosed with HIV or AIDS, they are given a NASCOP identification number (NASCOP ID), Yellow Appointment Card, and a comprehensive care card (NASCOP Blue Card) is started (Personal Communication Steven Waynee, 7.Dec.2010). While the NASCOP ID is unique to the individual clinic, it is not unique throughout the country (35). The Yellow Appointment Card collects various data points such as the patient's: demographic data, ARV information, appointment data info, lab diagnostics, and weight.

This card is given to the patients and should be brought back when the patients come for their next visit. The NASCOP Blue Card, Appendix A, identifies the minimum data set for HIV treatment and care. This card stays in the clinic and is updated each time the patient visits (36). These minimum data

elements include, but are not limited to: date of HIV diagnosis, ART treatment history and prophylaxis usage.

While the practice of maintaining both the appointment card and the NASCOP Blue Card allows both the patient and the healthcare facility to have access to the patient's HIV related medical information, there are challenges associated with this practice. These challenges include:

- The transcription of data between both cards, which can result in incorrect or incomplete data.
- The patient may not bring the Yellow Appointment Card to every medical visit, (again resulting in incomplete data).
- Information on either card does not provide data on non-HIV/AIDS related visits, (further adding to the problem of segmented health records).

The number and close proximity of health clinics, indicated by blue dots in Figure 3, above, and the tiered healthcare system within Kenya do not guarantee that a patient will visit the same health care facility each time they seek medical care, which decreases the ability of physicians to maintain continuity of care.

In addition to the NASCOP Blue Card, the Kenya Bureau of Standards (KEBS), in accordance with the International Standards Organization (ISO), developed standards for Health Information in Kenya. I-TECH, a joint effort between the University of Washington and the University of California San Francisco, the MOH, KEBS and other organizations in Kenya, has created a report called Standards and Guidelines for Electronic Medical Record Systems in Kenya (4). While the report has been completed, the work to implement the included recommendations is still underway. Until those recommendations have been implemented, data integration in Kenya will be difficult given the number of homegrown systems and lack of standardization. The minimum HIV data standards as well as the NASCOP Blue Card that were used to develop the HIV ontology were also used in creating the recommendations established in this report. Moreover, the ontologies can be modified to accommodate changes in current and future standards. The ontology-based approach presented in this dissertation can be of use in

integrating current systems and providing standardization until, and well after these recommendations have been fully adopted.

## 2.6. Terminologies and Ontologies in Medicine

In this dissertation, I will explore the use of ontologies as a means for facilitating data integration across disparate healthcare facilities. While XML provides syntactic structure to data in the medical domain, controlled vocabularies or terminologies and ontologies are used to standardize data within and between systems. XML also provides structure and is used by standards, e.g., HL7. A vocabulary or terminology defines the legal values associated with a particular domain and “are often associated with data entry” (37), (e.g., gender can be male or female). Examples of medical terminologies are: ICD-10, and Logical Observational Identifiers Names and Codes (LOINC).

Ontologies can be used instead of, or in addition to, terminologies. An ontology is a “formal explicit description of concepts in a domain of discourse” (38). The terms: class, property, attribute, and restriction are used to describe ontologies. Classes are entities or things in the domain, which are further described by properties. Properties define how classes relate to each other. Properties can also have restrictions, which limit the scope of the property. Restrictions can limit the property to only integers or to a specific type, a bed can be empty or not empty. (e.g., hospital = concept; bed = property; attribute = inpatient, outpatient; restriction = empty, not empty)

SNOMED CT, National Cancer Institute (NCI) Thesaurus, Unified Medical Language System (UMLS), the Foundational Model of Anatomy (FMA), the Ontology of Clinical Research (OCRe), and the Infectious Disease Ontology (IDO) are examples of ontologies used in medicine.

Ontologies are often represented using the Web Ontology Language (OWL). OWL “is a semantic markup language for publishing and sharing ontologies on the World Wide Web” (39). OWL, which is built upon the Resource Description Framework (RDF), is represented in XML format. The XML provides the structure, which allows the ontology to be processed and understood by a computer. The ontology, which is described using XML, provides the semantics, or meaning, of what is being represented. Semantics provide knowledge as to how pieces of information relate to each other. In the case of a

hospital and a bed, semantics would detail that a bed exists within a hospital room, or that a patient occupies a bed in a hospital.

Ontologies also define relationships between classes. These relationships help to provide intelligent querying--an important benefit of ontologies. Intelligent querying allows the user to ask questions of the ontology that are not explicitly defined. However, because intelligent querying is not always the main goal for developing medical ontologies, this functionality may not always be available, thereby limiting the usefulness of the ontology. Heja, et al found that because of ontological errors in SNOMED CT, automatic reasoning is a challenge (40).

These terminologies and ontologies cover a myriad of medical topics, but may not be sufficient to cover the types of data and information represented in clinical research settings of developing countries. Ontologies have been used in EMRs and health information systems to provide decision making capabilities, chronic disease management and data standardization (41–44). Richesson, et al, examined SNOMED CT to determine whether it would be sufficient to cover the concepts used in the Arthritis, Rheumatism, and Aging Medical Information Systems. They concluded that SNOMED CT provided some coverage, but was not sufficient to cover all of the concepts for their domain (45).

Besides developing ontologies to provide standard nomenclature throughout and across domains, researchers have created systems that use ontologies to describe clinical trials and their results (46–49). Ontologies have also been used to automatically create laboratory information systems to support those trials (50). For example, the ontologies created by Das and Li are sufficient to describe the components of a clinical trial (study arms, randomization, study visits), but not the types of data that are collected (blood pressure, weight, and BMI) (18, 22).

While Das and Li are working to semantically describe the management of clinical trials, The Human Studyome project is working to create a standardized system for describing human studies data, trial structure and design, as well as outcomes, study documents and eventually participant-level data (51). The Human Studyome Project led by Id Sim, which includes OCRe, takes her prior work, Trial Bank, one step further. Trial Bank is a project that works to standardize the way clinical trial results are reported, using trial bank databases mapped to a shared clinical trials ontology (52). These projects, while integral to the task of sharing data about clinical trials and their results, are not meant to sufficiently

describe the types of data and information created and used by clinicians and researchers in low resource settings for the treatment and care of HIV/AIDS patients.

These concepts are not unique and have been represented in various terminologies and ontologies; however, not in a single comprehensive ontology. I will utilize concepts from the ontologies and terminologies discussed above to supplement the HIV/AIDS treatment and care and Patient Identifier ontologies that will result from this work. The use of these ontologies will provide concepts and language consistent with those currently being used in the medical community and aid implementation should users decide to employ either of these in addition to our ontologies.

## **2.7. Unique data in this setting**

While the HIV/AIDS treatment and care ontology will contain classes that are familiar to the medical community as a whole, classes will be added to provide for differences in the way treatment is managed in low-resources settings. The biggest differences between these settings will be realized in the Patient Identifier ontology. Data points, such as name and age, are applicable in both developed and developing countries; however, low-resource settings present a challenge when it comes to concepts such as date of birth and location. Date of birth will be broken into its component pieces (month, day, year). Inhabitants in rural areas of Kenya often do not know their exact month or day of birth-- in some cases, only the year. This is in stark contrast to citizens of the United States and other developed countries.

Location is also important, as it is a critical component of the Kenyan equivalent of an address in the United States. In the United States, street address, city, state and zip code make up a complete address. In Kenya, the nearest health care facility or landmark is also used to identify a patient's "address." This is especially important when street names and/or numbers are not present (53). Address and date of birth are just two ways in which patient identifying information is different in Kenya. Table 1 below provides a list of a few differences in identifying patients in the U.S. as compared to Kenya.

USA	Kenya
Birthdate: Known	Birthdate: Known/Unknown
Standard Address	Non-Standard Address/ Location
Social Security Number given at birth	National Social Security Fund number given upon employment
Marital Status: Married, Single, Divorced, Widowed	Marital Status: Married Polygamous, Married Monogamous, Divorced/Separated, Widowed, Cohabiting, Single

**Table 1 - USA vs. Kenya Identifying Information Differences**

At birth or shortly thereafter, in the United States, a Social Security Number (SSN) is issued to every citizen. Unfortunately, valid SSNs are not unique, and alone they are not suitable to identify the owner. However, if other pieces of identifying information were used in addition to the SSN, the owner can be identified. The Kenyan “equivalent” to the U.S.’s SSN is the National Social Security Fund number. However, using this as a national identification number is not possible because it is only issued upon employment. Finally, marital status also presents a difference from that of the U.S. In the U.S., there are generally three options: 1) Married, 2) Single, and 3) Divorced. Extra options such as Married Polygamous and Married Monogamous (provided in Kenya) are not legal or socially acceptable in the United States. However, monitoring prevalent relationships is key to understanding and controlling HIV in the population.

## 2.8. Record Linkage

In the absence of a unique patient identifier (UPI) or a Master Patient Index (MPI), linking patients between two systems can be difficult. Record linkage or patient matching is the task of accurately labeling record pairs corresponding to the same sources (54). There are two types of record linkage: deterministic and probabilistic. Deterministic record linkage looks “for exact (dis)agreement on one or more matching variables between files” (55). Probabilistic methods use “information on a greater number of matching variables and allows for the amount of information provided by any (dis)agreement on matching variables” (55).

The United States has no UPI and currently uses statistical matching to identify patients (56). Probabilistic matching has proven to be effective and provides high sensitivity and specificity (57,58). However, in Kenya this method is problematic in that personal attribute keys (name, age, date of birth, address) are not “unique to the individual, change over time and are often entered into different systems in different formats. Data-entry errors, such as misspellings, add to the difficulties with this type of key” (56). The Expectation-Maximization (54) and Fellegi-Sunter (54,59) algorithms are used often in the medical field to provide probabilistic record matching.

In addition to a lack of a MPI, or SSN such as in the United States, names are also used to facilitate matching. The Soundex algorithm was created for this type of matching (60). The algorithm is based on an English pronunciation of names to phonetically match them. Although this algorithm has been used throughout the United States and the developed world, it has proven to be insufficient for use in many cases because, it has a dependence on initial letters, noise intolerance, different transcription systems, and silent consonants, to name a few (61,62). More importantly, it is insufficient for applications related to Bantu languages which includes Swahili, the official language of Kenya (63).

The inadequacy of Soundex can be further compounded by the lack of emphasis placed on spelling names correctly in Kenya-- instead correct pronunciation is highlighted. Moreover, certain tribes in Kenya such as the Kamba and Kikuyu tribes substitute the letter “L” for “R” and “R” for “L”, respectively (Personal Communication Edwin Wambua, October 2012). The “L”/“R” problem has been addressed by the New York State Identification and Intelligence System (64) and Metaphone (65) algorithms; however, again it has been found that these algorithms are not sensitive enough to cover African dialects (66).

Both methods, deterministic and probabilistic, rely on the appropriate mapping between systems to provide an accurate match. Mapping between systems is usually done manually and is often facilitated by the use of terminologies and ontologies.

## **2.9. Why ontologies in Kenya?**

As I have shown, there are three major challenges in data management:

- 1) understanding the types of data that need to be stored, specifically those data points that are different from that of the U.S.,

- 2) integrating new systems that have standards, and,
- 3) integrating the systems that are currently deployed that may or may not be standard-driven.

These challenges are the focus of the MOH's work to implement standards for EMRs, which will aid in data integration. While the MOH's work will provide a clear answer to the first two challenges, the third continues to be a challenge. I believe ontologies will aid the systems without the MOH's standards to exchange data with the systems that were built based on those standards. Standards such as the NASCOP Blue Card, as well as national and international level standards that are currently being used, should be the basis for these ontologies. Once MOH's work has been completed, these ontologies can be used in addition to or incorporated with the MOH's standards. More importantly, the quality of patient care will depend on access to data in newly adopted and legacy systems within a facility as well as data that can be found in systems outside of said facility. Facility level developers of legacy systems and clinicians could lead the effort in development and maintenance of these ontologies. These developers and clinicians will have first hand knowledge of the data necessary to provide patient care and of what the legacy system's capabilities are in terms of storing and managing data.

## **2.10. Data Integration**

Data integration provides the ability to query across data sources. These sources can be either homogeneous or heterogeneous. There are four data integration architectures: data warehouses, database federations, database federations with mediated schemas and distributed query systems (67). The differences in these systems is based primarily on where the data are stored, how up-to-date the data are at any given time and how much human interaction with the data is required.

In data warehouses, the data are stored in a central database. The data are entered into the system through direct input, via an always-on or intermittent connection to the source or by the source sending the data by manual means. An advantage and disadvantage of the system is that there is "usually a high amount of human 'interaction' with the data, often in the form of quality control and curation." The data housed in a data warehouse may not always be up-to-date (67).

Federated databases store data locally at the source, instead of centrally (data warehouse). Mapping between sources is not necessary because the same schema is used for all data sources. Human interaction with the data is limited as curators are usually not involved. Data in a federation is always up-to-date since it is kept at the source (67).

Database federations with a mediated approach maintain the premise that the data is at the source, but each source is in turn mapped to a single schema that applies to the entire federation. This allows the researcher to understand one schema, and obtain answers from multiple sources instead of learning multiple schemas. Like the database federation approach, the data are always up-to-date (67).

An example of this approach is the BioMediator system. Built by Dr. Peter Tarczy-Hornoch's Biomediator research group at the University of Washington, Biomediator was created to provide biologists with "a data integration system that provides a common interface to Web-accessible sources of biologic information (68)." The Biomediator architecture consists of "six components: source knowledgebase (SKB), query processor, metawrapper, plugin, wrapper and data sources (68)."

The Biomediator source knowledge base (SKB), represented in Protégé and accessed via the Protégé API (application programming interface), holds the mediated schema (ontology), mediated schema annotations, all data sources and the elements from the mediated schema about those sources (68). The Query Processor, accessed through its API, manages queries asked using the mediated schema. The user using the programming query language (PQL) submits these queries (69). These queries are then translated into queries over the source databases using the metawrapper. The plugin connects the data access portion (wrapper, metawrapper, data sources) of Biomediator to the rest of the system (SKB, query processor). These queries are then sent to the source database using the wrapper. Results are returned from source databases and transformed into an XML document and output to the user.

The prototype system that will result from my dissertation work will use aspects of the Biomediator system -- specifically, the metawrapper idea, in which queries are translated based on the user's query into queries that are in a format accessible by the source database.

The final architecture, distributed query systems, allows the user to query data sources that are not stored centrally (data warehouse). Query Integrator (QI), a query management system developed by

the Structural Informatics Group (SIG) at the University of Washington, implements a distributed query approach. QI, previously called Query Manager, “provides an interface and framework for editing, executing, storing, and discovering views” (70). The QI architecture consists of core components and coordinated web services. The Core Components are: the Query Database, QI Server and the Query Execution Service (QES).

An important component of QI is its work with web services. Queries input by the user in an Adobe Flex client user interface (UI), are stored in a PostgreSQL database, the Query Database, and given an ID. The ID as a part of a uniform resource locator (URL) is used to execute the query using the QES, a representational state transfer (REST) web service (71). This allows the QI to execute single queries as well as chained queries, one query referring to a URL from another query. The results are returned to the user in XML format. The QI server facilitates the interactions between the UI, core components and the coordinated web services. These web services are query engines for the “following languages: XQuery, DXQuery (distributed XQuery), SPARQL Protocol and RDF Query Language (SPARQL), vSPARQL (view SPARQL), IML (immediate language), and two convenience ‘languages’”(71).

Web services like those used in QI, in the form of APIs, will also be explored in developing the prototype system. This will allow users to access data sources through protocols developed by the owner of the data source. APIs have built in logic, which could reduce the workload on the prototype system, by reducing the amount of translation needed to create queries specifically for the data source to which the API belongs.

The prototype system I have developed implements the database federation with a mediated schema architecture as it complements the current clinical information system landscape in Kenya. It will allow organizations to use their existing and/or implement new information systems, standards-based or not, to gain access to multiple heterogeneous systems. In this case, the ontology will act as the mediated schema.

## **2.11. Ontologies can aid in data integration**

Traditionally, there are three approaches for ontology use in facilitating data integration: single

ontology, multiple ontologies, and hybrid ontologies (72). A single ontology or “mediated schema” is mapped to all data sources. The multiple ontology approach associates a separate ontology for each data source. Alternatively, the hybrid approach also provides one ontology for each data source; however, the ontologies are created using a shared vocabulary (72). This work will utilize the single ontology approach.

While each of these approaches has its own associated strengths and weaknesses, the single ontology approach will be used because of its simplicity. This approach eliminates the need for the user to understand multiple database schemas. Instead, the single ontology approach “provides a mechanism to define queries based on the concepts of the ontology and present the query results in a unified and structured form” (73). Additionally, a global ontology eliminates the need for organizations to maintain their own ontology.

Ontologies are often used to create open-source software for data integration. I2b2, informatics for integrating biology and the bedside, is an open-source system built by the Partners HealthCare System, to integrate EMR data with clinical research data. An implementation is called a hive. Cells within the hive “store data or contain analysis methods that facilitate the repurposing of medical record data for research” (74). Each patient observation or data point is mapped to an ontology cell, which helps to facilitate querying of the system. In addition to i2b2, other researchers have used ontologies for data integration (73,75,76).

It is important to note that each of these systems has used ontologies to integrate multiple systems, not just two. Mapping two systems to each other is not sufficient justification to employ an ontology. Without an ontology, the addition of each new system requires a pairwise mapping to each previous system per concept, as compared to just one mapping to an ontology. Equation 2.1 below shows the number of between system mappings per concept without an ontology where  $N = \#$  of systems.  $(N-1)$  is the number of systems to which one system has to map. Dividing by two ensures that mappings are not counted twice. A visual example of this can be seen in Figure 4 below. Integrating four systems without an ontology requires six pairwise mappings per concept. Integrating four systems using an ontology only requires that each individual system be mapped to the ontology once per concept.

$$\# \text{ between system mappings per concept} = \frac{N * (N - 1)}{2} \quad (2.1)$$

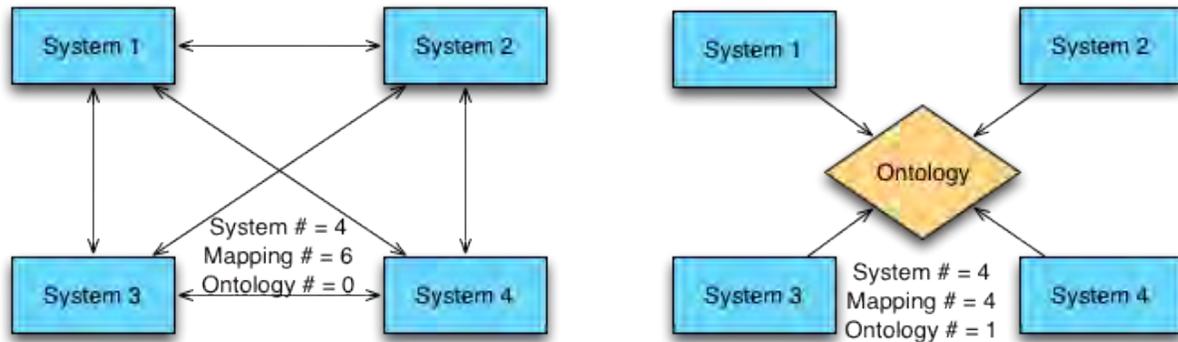


Figure 4 - Pairwise Mapping vs. Ontology Mapping Per Concept

## 2.12. Ontology-Based Data Integration Approach

This research will implement the database federation with mediated schema architecture approach. The ontology will be developed using the Protégé system in the OWL 2 language.

## 2.13. Conclusions

In this chapter, I have discussed HIV in Kenya and the need for sharing data across separately maintained systems in order to reduce redundancy and to provide the maximum amount of information for clinical care and research (77). To do this requires data standards and a data integration architecture, which are sparsely used in Kenya. Next, I discussed information systems and their increasing popularity, especially homegrown and open-source systems. Unfortunately, these systems are siloed within their own departments and are often used in addition to paper-based processes. Finally, I discussed how ontologies are useful given the lack of data standards, disparate data sources and lack of data integration in low-resource settings. The work in this dissertation addresses these challenges by providing a patient identifier and an HIV/AIDS ontology that incorporate data models and systems, OpenMRS and OpenClinica, that are currently in use in low-resource settings.

## Chapter 3: Aim 1 – Describing Clinical Data

As a result of the disparate purposes, designs, and architectures of information systems, data integration can be difficult to achieve in low-resource settings. Furthermore, system developers are confronted by an array of legacy functional needs, clinical aims (treatment, prevention, and clinical trials), and both official and ad hoc standards. To address this diversity of both information uses and data models, I describe the current level of standardization of patient identifiers needed to link data between systems. Next, I introduce a prototype patient identifier ontology, which can be used to aid in the data integration of heterogeneous clinical information systems (CIS). I then repeat this process as I seek to identify the types of data and information used in the treatment and care of HIV/AIDS patients. Finally, I develop a prototype HIV/AIDS treatment and care ontology, which will be used along with the patient identifier ontology to create an OBDIS for open source EMR and EDC systems in Chapter 4.

Establishing standards in an environment with existing competing data models can be difficult. It is important first to establish the level of agreement between data representations and the overlap in domain coverage. In a prior study of identifying information used in TB contact investigation forms in fifty states and three countries, Abernethy et. al. (78) revealed a broad range of data field frequency used to identify patient contacts. In the setting of fragmented HIV care systems and competing guidelines, the current state of standardization particular to this context must be assessed.

### 3.1 Data Collection

The methods for this research were derived from (79). The work described in this chapter is an expansion of the work described by Guidry, et al (80). A convenience sample of fourteen data models, information systems, and standards utilizing patient identifiers were obtained: the RadLex (81) ontology; the OpenMRS (25) and OpenClinica (82) database schemas; a peer-reviewed paper on the Mosoriot Medical Records System (MMRS) (83); a report by RAND Health on Unique Patient Identifiers (84); the Kenya National AIDS/STD Control Programme Comprehensive Care Patient Card (NASCOP) Blue Card (85); WHO patient monitoring guidelines (86); five HIV Care/ART Cards from Namibia (87), Uganda (88), Tanzania (89), and WHO European and Southeast Asia Regional Offices (SERO) (86); Centers for

Disease Control and Prevention – Council of State and Territorial Epidemiologists (CDC-CSTE) (90); and the Johns Hopkins Patient Identification System website (91).

Technical information on data models was obtained from online manuals, published descriptions, reference websites, and personal communication with system developers. In order to include an example ontology in the sample, I searched the BioPortal (92) website which facilitates the sharing of ontologies in the biomedical community. A search using the keywords “patient identifier” returned three ontologies: RadLex, RadLex in OWL, and LOINC. LOINC was not included in this study due to its specialized focus on laboratory results. RadLex and RadLex in OWL’s patient identifying information were identical; hence, they were treated as a single source for our purposes. RadLex provided six patient identifier fields.

In Kenya, patients enrolled in government-funded health care clinics have data recorded on a card provided through NASCOP, an agency of the Kenya Ministry of Health. The “Blue Card” standardizes the demographic and treatment information collected on each patient. This card remains in the patient’s file at the clinic and is updated with recent information during each clinic visit. While this keeps the patient’s information in the same place, it is often not in computable format, because the majority of health care facilities in Kenya do not have a CIS. Therefore, most of this information is either kept on paper forms (internal clinical forms) and/or the “Blue Card.”

The NASCOP Blue Card is often compared to the WHO patient monitoring guidelines for HIV. This dataset is used throughout the world for the treatment of HIV/AIDS and is a part of the WHO standard for EMRs. The HIV Care/ART Cards used in this study are much like the NASCOP “Blue Card” and the WHO guidelines. However, because of regional differences in treatment protocols, each form has data points that differ from those of their peers.

The forms for Uganda and Tanzania were found on the International Epidemiologic Databases to Evaluate AIDS – East Africa Joomla website via a Google search for “HIV Care and ART Card.” Likewise, Namibia’s card was located in the Namibia Patient Care Booklet on the AIDStarOne site via the same Google search parameters. Finally, the WHO Euro and SERO cards were included as examples in the WHO Patient Guidelines document.

In addition to the WHO standard system, the Centers for Disease Control and Prevention have created the CDC-CSTE dataset to aid in the standardization of information requested by the CDC and other public health agencies related to disease surveillance.

The peer-reviewed paper (MMRS) and report (RAND) were found using a Google Scholar search for patient identifier information. The MMRS paper specifically discusses the patient identifier information needs related to implementing an EMR in Kenya. A discussion of what patient information would be needed to implement a unique patient identifier for the United States is detailed in the RAND report. Finally, the Johns Hopkins Website discusses their Patient Identification System. This system assigns medical record numbers to patients based on their personal identifying information and facilitates merging the medical records of patients having duplicate record numbers.

Table 2 below provides a list of each data source, its type and primary user.

<b>Data Source</b>	<b>Type</b>	<b>Primary User</b>
RadLex	Ontology	Used by radiologists
OpenMRS	Database schema	Used to capture observations from clinic visits
OpenClinica	Database schema	Used to capture clinical research data, worldwide
MMRS Paper	Describes EMR System	Used in Kenya
RAND Report	Describes the procedures used to identify patients	USA
NASCOP “Blue Card”	HIV Care/ART Card	Used throughout Kenya by all government-funded medical facilities
WHO Dataset	Recommendations by the WHO for HIV/AIDS care	Used throughout the world
Namibia – HIV Care/ART Card	HIV Care/ART Card	Namibia
Uganda – HIV Care/ART Card	HIV Care/ART Card	Uganda
Tanzania – HIV Care/ART Card	HIV Care/ART Card	Tanzania
EURO – HIV Care/ART Card	HIV Care/ART Card	WHO Euro Region
SERO – HIV Care/ART Card	HIV Care/ART Card	WHO SERO Region
CDC-CSTE Dataset	Recommendation by the CDC-CSTE Working Group	Used to standardize infectious disease investigation
Johns Hopkins Website	Describes a Patient Identification System	Used by Johns Hopkins Medicine to identify patients in disparate clinical information systems

**Table 2 - Data Sources, their type and primary user**

Composition of the data models was tabulated in a Microsoft Excel spreadsheet (Table 3). The rows of the spreadsheet are the variable or column names from the article, database schema, ontology or

resource. The columns of the spreadsheet were: the names of the system, ontology or resource described. If a system or ontology used a particular variable or column name to identify patients an 'x' was placed in the box corresponding to that column and row.

## **3.2 Patient Identifier Ontology**

This resulted in forty-one data fields, of which twenty were unique to one model. The completed data collection table can be found in Appendix B. Data fields were organized according to their semantic content into the following categories: Numeric identifiers, Patient information, Relative information, and Location. Table 4 below provides a list of these data fields and examples of each from the data model survey. These categories subsequently became classes or properties in the ontology.

	RadLex	OpenMRS	OpenClinica	WHO - Minimum HIV/AIDS DataSet	MMRS	NASCOP	CDC-CSTE	Johns Hopkins	RAND Health	Uganda - HIV Care/ART Card	EURO - HIV care/ART Card	SERO - HIV care/ART	Namibia - HIV Care/ART Card	Tanzania - HIV Care/ART Card
Medical Record Number	x	x												
StudyID			x											
Patient Age	x			x		x	x			x		x	x	x
Patient Date of Birth	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Patient Ethnicity	x						x							
Patient Gender	x	x	x	x		x	x	x		x	x	x	x	x
Patient First Name		x		x	x		x	x	x					
Patient Middle Name		x			x		x	x	x					
Patient Last Name				x	x		x	x	x					
Patient's Mother's First Name					x									
Patient's Home Village		x			x									
Patient Name	x	x				x				x	x	x	x	x
ClinicID				x						x	x	x	x	x
Unique Patient Identifier				x		x				x	x	x	x	x

Table 3 – Patient Identifier Data Model Survey – Abbreviated

<b>Categories</b>	<b>Examples</b>
Numeric Identifiers	Medical Record Number; ClinicID; Unique Patient Identifier
Patient Information	Age; Date of Birth; Name
Relative Information	Mother's First Name; Next of Kin; Next of Kin Telephone
Location	District; Nearest Landmark; Nearest Health Facility

**Table 4 - Patient Identifier Ontology Data Field Categories and Examples**

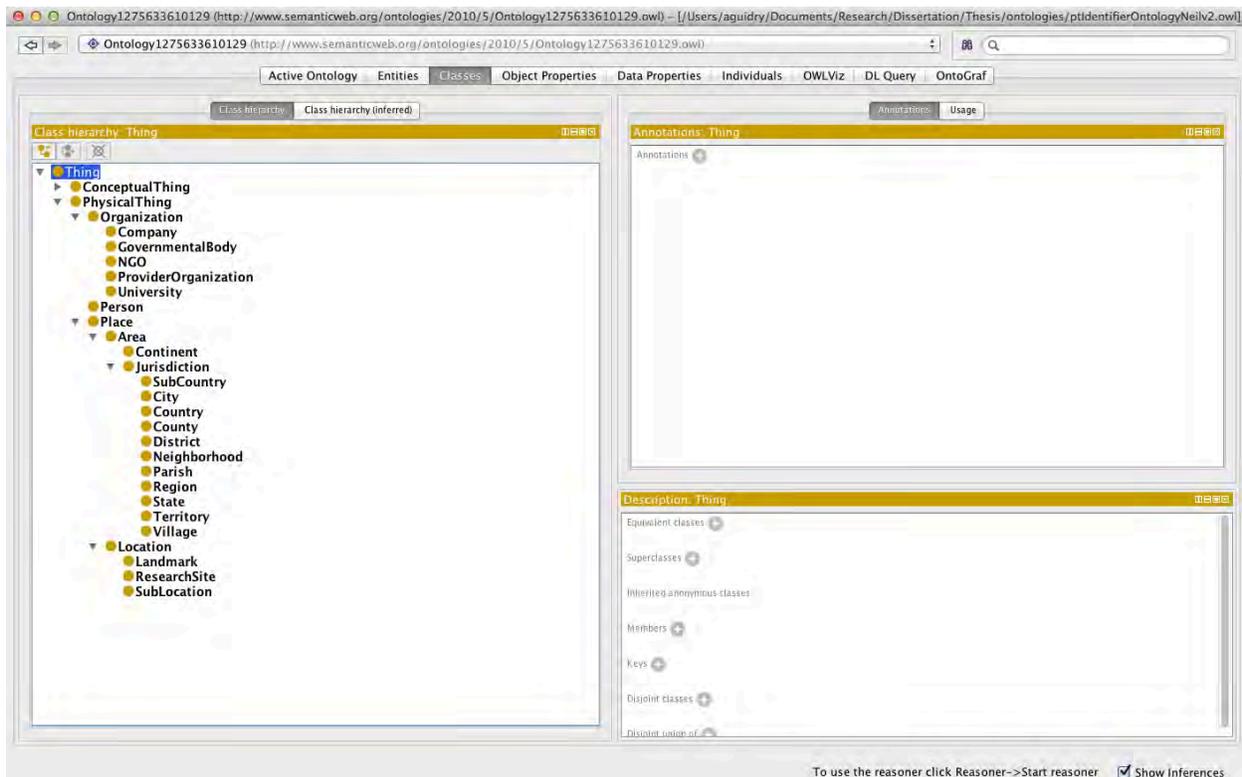


Figure 5 - Protege 4.1 Screenshot

Data from the spreadsheet was used to create a patient identifier ontology using the Protégé 4.1 (93) system (Figure 5). We created an OWL-based ontology having fifty-two classes, eighty-four properties (63 object and 21 data), and nine individuals using the bottom-up ontology development approach, starting with concepts from the data model summary classes as seeds. Object Properties define relationships between two classes or instances of classes (e.g. (hasAddress Person Address)). Person and Address are both classes linked by the object property hasAddress. Data properties link a class to a value (e.g. (hasAge Person 8)). Individuals are instances of classes (e.g. Male, Female of class Gender). The number of classes outnumbers the source fields due to generalization of the data model to accommodate broader data types and instances. *Patient* and *Provider* both generalize to the class *Person*, each of whom will have some distinct and shared properties.

Properties (also known as slots) describe the features of class members. For example, each *Person* (including a *Patient*) has a *Gender* and a *Name*. The *hasGender* property links an instance of

*Person* to an instance of the class *Gender*, such as in the assertion “(hasGender Tony Male).” Most properties in our ontology correspond directly to the data fields compiled in the rows of Table 4.

### 3.2.1 Data Model Analysis

Of the forty-one unique data fields, only *date of birth* appeared in all sources. However, use of this field may result in misclassification bias, as exact day, month and year of birth is unknown in certain areas of the world (83). Nine used *telephone* while, *age*, *patient name*, and *marital status* appeared in eight data sources. Among data categories, the most common numeric identifiers were *ClinicID* and *Unique Patient Identifier*. The most common patient information fields were *date of birth* and *gender*; however, all data models included some version of *Patient Name*. The most common location information fields were *Street address*, *Postal code*, *District*, and *Telephone*. Data fields pertaining to patient relatives were only used in three data models in our sample, and no field occurred more than once.

Comparing data models, the CDC-CSTE, OpenMRS, WHO HIV/AIDS minimum dataset, and the HIV Care/ART Cards used similar collections of location data. Among patient information data fields, similarities are seen between the RadLex and NASCOP models (which depend heavily on *age* and *gender*), and between MMRS, OpenMRS and Rand Health (which specify *first name* and *middle name* with few other fields). This likely reflects the derivation of OpenMRS from the MMRS system. The HIV Care/ART Cards provide a wide range of data fields, which span all data categories. Other similarities exist between systems but are less easily grouped into meaningful sets.

### 3.2.2 Ontology Design

Building on the results of the data model analysis, we have created a prototype patient identifier ontology (PIDO) to capture the explicit semantics of each data field in a formal, computable description. The class hierarchy of this ontology can be seen in Figure 6 and Figure 7.



Figure 6 - Patient Identifier Ontology - Conceptual Thing Class Hierarchy



**Figure 7 - Patient Identifier Ontology - Physical Thing Class Hierarchy**

As Figures 6 & 7 illustrate, the Patient Identifier Ontology describes information about the patient unrelated to his or her medical history. Specifically, this information would be reflected in a typical medical enrollment form, under the heading of “Personal Information.” While individually these data fields may not be able to identify a specific patient, together they provide enough information to not only identify a patient, but also indicate whether a duplicate patient exists as well.

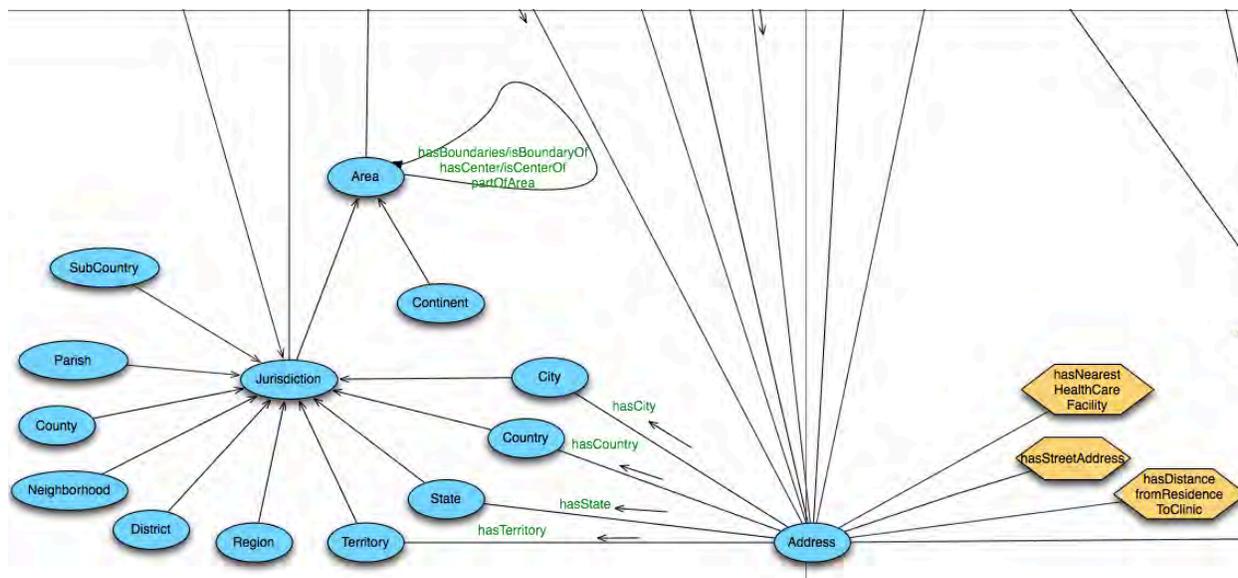
The two top-level classes of the ontology are: *Conceptual Thing* and *Physical Thing*. They each have subclasses, which specialize the superclass. *Conceptual Thing* (Figure 6 above) includes the subclasses: *Attribute*, *DataCollections*, *Identifier*, *ResearchStudy* and *Role*. *Attribute* is specialized by the *Ethnicity*, *Gender*, *MaritalStatus* and *Race* subclasses. The nine individuals used in this ontology are instances of the *Gender* (e.g. *Male*, *Female*) and *MaritalStatus* (e.g. *Single*, *Married*) classes.

The *Role* class identifies the functions a person can perform. Specifically, this class describes the types of roles, which interact with a *Patient* (e.g. *Treatment Supporter*, *Provider*) as well as the *Patient* himself. A *Patient* can also function as a *Research Study* participant. While these functions are distinct, they are not mutually exclusive and a person can function in one or all of these roles.

The *Conceptual Thing* class also includes the *Address*, *ContactNumber* and *ClinicalStudy* classes.

The *PhysicalThing* class includes the top-level classes: *Organization*, *Person* and *Place*. The *Organization* class is specialized by classes that describe funding and medical care agencies. The *Person* class does not have a specialization, but is connected to the other classes by object properties. Moreover, data properties are also used to provide information about a person.

Finally, the majority of the *PhysicalThing* class is centered around the *Place* subclass. This class includes the *Area* and *Location* subclasses. These classes are representative of what an address in a low-resource setting would include, specifically e.g. *Location*, *Nearest Health Facility*.



**Figure 8 - Visual Diagram of Jurisdiction and Address classes from the Patient Identifier Ontology**

Figure 8 above, provides a visual representation of the *Jurisdiction* and *Address* classes and subclasses, a complete visual representation of the ontology can be seen in Appendix C. Classes are indicated by ovals (blue), data properties by octagons (goldenrod), and object properties (green) by text near lines and directional arrows. The data properties represent data fields that are members of the “Location Information” category.

The *Area* class has object properties however, they lack directional arrows. This indicates that the domain and range of these properties is the *Area* class itself.

*Jurisdiction* has eleven subclasses, which represent many of the ways an area of land can be divided. While most of these are not used when representing address and/or location in the western world, in other parts of the world these divisions along with *Location*, *Sub Location*, *Landmark* and *Nearest Health Facility* are integral components of *Address*.

### 3.3 HIV Treatment and Care Ontology

Using the same methods as those in the Data Collection section (Section 3.1), I analyzed seven of the fourteen data models from Table 2 above. Only the NASCOP “Blue Card,” WHO Dataset and the

HIV Care/ART Cards from Uganda, WHO Euro and SERO Regional Offices, Namibia and Tanzania were used. The other data sources were excluded from this study because they contained patient identifier information only.

This resulted in two hundred data fields, of which sixty-four were unique to one data model. The completed data collection table is included in Appendix F. Again, the data fields were organized according to their semantic content into the following categories: Patient Information, HIV Care and Family Status, ART Summary, Outpatient Encounter-Level Information and Counseling. There were data points that fell into the Patient Information category, but were omitted because they were unrelated to HIV Care and were only used for the Patient Identifier Ontology. Table 5 below provides an example of the data fields that comprise each category. The HIV Ontology (HIVO) has 28 classes, and 286 properties (24 object and 262 data).

<b>Categories</b>	<b>Examples</b>
HIV Care and Family Status	Name of Treatment Supporter; Entry Point into HIV care; Child/partner/family member HIV Status
ART Summary	Date determined medically eligible to start ART; ART cohort; First ART regimen at this facility
Outpatient Encounter-Level Information	Next scheduled outpatient visit date; Visit Type; WHO Clinical Stage
Counseling	Why complete adherence needed; Explain dose, when to take; Treatment supporter preparation

**Table 5 - HIV Ontology Data Field Categories and Examples**

### 3.3.1 Data Model Analysis

Of the 200 data fields, 36 fields were present in all data sources, most of which were in the Patient Information and ART Summary categories. The completed data tables for the HIVO can be found in Appendix F. Sixteen fields were present in six data sources and fifty-five fields in five sources. The overlap in fields can be credited to the use of the WHO Minimum Data Set as a model for developing HIV Care/ART Cards.

The 64 unique fields can be attributed to regional differences in treatment protocols. For example, the NASCOP card was the only card that included malaria status. The WHO SERO card provided many data points unlike those of other data sources specifically, socio-economic status of the patient and mode of HIV transmission. Finally, the WHO Euro card was the only card to include information about the patient's Hepatitis B and C status.

### 3.3.2 Ontology Design

The scope of the HIVO was to describe the types of data created and used in the treatment and care of HIV/AIDS patients. This includes the following clinical purposes: disease state, symptoms, laboratory data, and medical and family history. It is not designed to describe the molecular biology of the virus or its epidemiology. To this end, the ontology is not limited to information about the patient's antiretroviral drug regimen. It also provides information about Pre-ARV counseling and family planning during treatment. Another example of this can be seen in the Disease class. In addition to the HIV class, TB (a common O.I.) is also included. While there are many O.I.s, TB was the only one used consistently across all data sources. However, because ontologies are intended to evolve, other instances of the disease class can be added as needed.

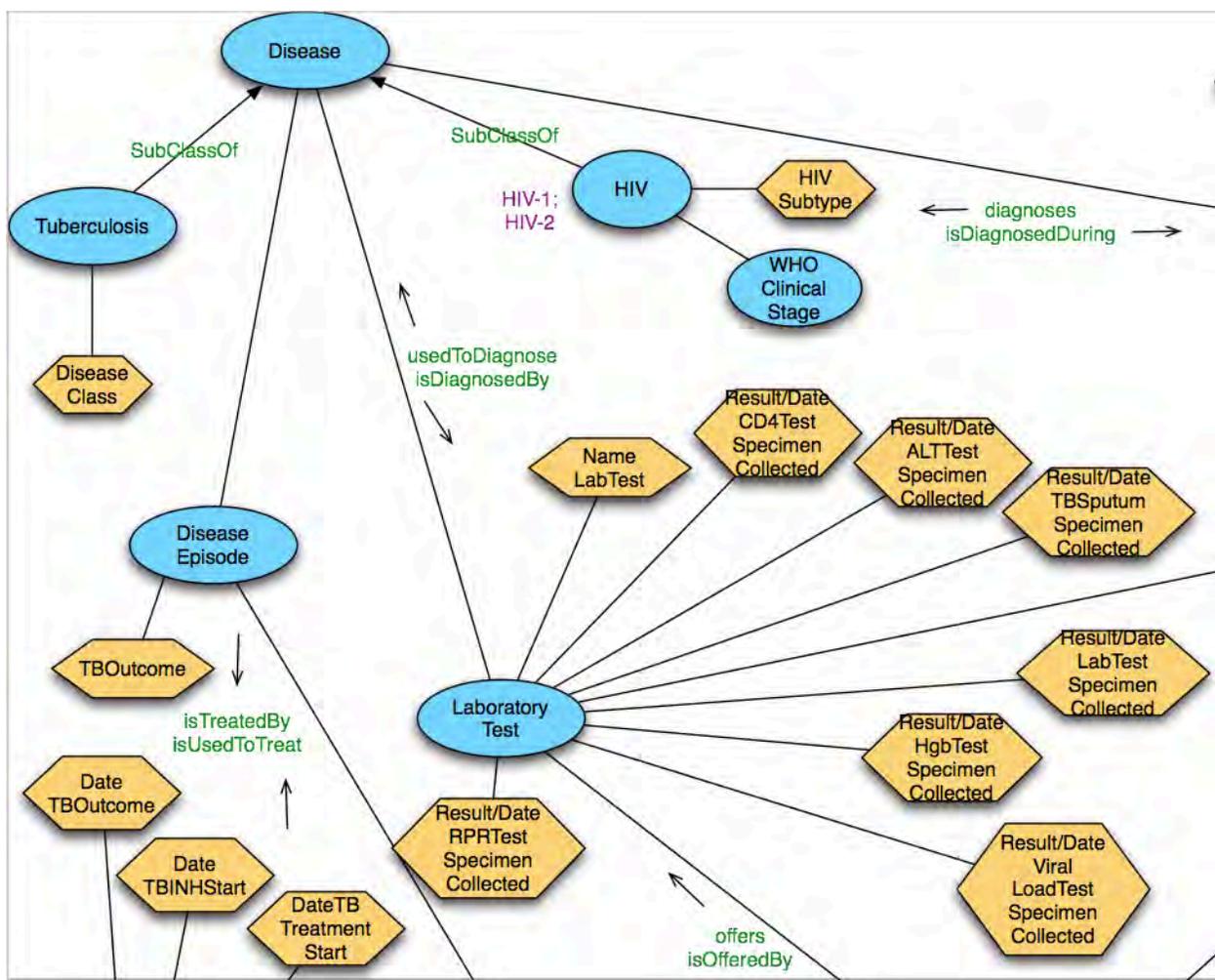
The class hierarchy of the resulting HIVO can be seen in Figure 9 below. Twelve top-level classes exist for this ontology: *Clinic*, *Disease*, *DiseaseEpisode*, *Drug*, *DrugAllergy*, *DrugRegimen*, *LaboratoryTest*, *MedicalHistory*, *Outpatient Encounter*, *Treatment*, *TreatmentEpisode* and *WHOClinicalStage*. The bulk of the ontology is focused around the Medical History class. This class has *ARVHistory*, *FamilyHistory* and *PersonalHistory* as subclasses. These subclasses represent data fields from the ART Summary, Outpatient Encounter-Level Information and Counseling categories.

Figure 10 below, shows the *Disease*, *Tuberculosis*, *DiseaseEpisode*, *LaboratoryTest*, *HIV* and *WHOClinicalStage* classes. Classes (blue) are represented by ovals and octagons are data properties (goldenrod). The *SubClassOf*, *isTreatedBy*, *isUsedToTreat*, *usedToDiagnose*, and *isDiagnosedBy* are object properties (green) that connect classes. The arrows near object properties identify their directionality, more specifically, their domain and range. Text directly above or below the data properties is what Protégé calls “Data Property Range Restrictions.” The restrictions (purple) mimic the options the user has on data entry forms. For example, the *HIV* Class has restrictions HIV-1 and HIV-2 to represent the sub-types of the HIV disease. In the instance of HIV, the “restrictions” are mutually exclusive; however, this is not always the case. This specific data field’s restrictions were the same across all data sources; however, in cases where the restrictions differed, all options are included. A full visual representation of this ontology can be found in Appendix G.



Figure 9 - HIV Ontology Class Hierarchy

The HIVO was designed to accommodate queries that are useful to both clinicians and researchers. Specific questions such as what medication is the patient on, what symptoms has the patient had since their last visit or what changes in the patient's ARV Drug Regimen have occurred since beginning ART can be asked of the ontology. Additionally, questions which are not specifically coded in the ontology can be asked. Example: *A physician wishes to identify all of his patients who presented with or were treated for malaria or tuberculosis in the past ten months.* In addition to returning a result set that contains all of the patients with a diagnosis of malaria or TB, the query will return a list of all patients who were prescribed medications or who presented with symptoms associated with malaria or TB.



**Figure 10 - Visual Diagram of Disease, Tuberculosis, Disease Episode, Laboratory Test, HIV and WHO Clinical Stage classes from HIV Ontology**

### 3.4 Conclusions

In this chapter, I have detailed the data collection process used to gather data sources, which describe the use of patient identifier and HIV data fields in treating HIV/AIDS patients in the USA and in low-resource settings. Finally, I have provided a patient identifier ontology, which can be used to facilitate data integration and exchange. However, this ontology alone is not enough to provide this functionality. Along with mapping the ontology through some sort of manual, automatic or semi-automatic means, one needs to determine whether the information returned is accurate (i.e. the patient being returned is the patient being searched for). This work alone will be used in Chapter 4 to implement a prototype OBDIS.

## Chapter 4: Aim 2 – Data Integration & Exchange

In the previous chapter, Chapter 3, I described the process by which I created the PIDO and HIVO, and the ontologies themselves. This chapter begins with a discussion of the requirements of an OBDIS in low-resource settings, followed by a description of the prototype system's architecture, and the technical details of the system.

Throughout this chapter, I will use the following four terms frequently:

- *Implementer* - someone who is installing, maintaining, or troubleshooting the system. This person may or may not interact with the system regularly, e.g. Information Technology (I.T.), database (DB) administrator, programmer/developer.
- *End user* - someone who interacts with the system regularly, e.g. data clerk, clinician, or researcher.
- *User* - an implementer or end user.
- *Data Source* - a health information system to be integrated, a database, e.g. CTMS, EMR, EDC.

### 4.1 Requirements and Scenarios of Use

During 2007 and 2008, I created a framework for characterizing EDC systems (29). With this work, I travelled to Nairobi, Kenya in 2009 to help Dr. Judd Walson and his study staff determine options for his next generation CTMS. At that time, he and his staff were using a PHP (Hypertext Processor)/MySQL/JavaScript based system to collect data for his Empiric Therapy of Helminth Co-Infection to Reduce HIV-1 Disease Progression (THE or PHE) (94) study. I also sat in on meetings related to the system and conducted informal interviews with his study staff to determine their information needs related to current and future clinical trials, and related projects.

Based on these information needs, various CTMS and EDC systems, programming languages and frameworks, as well as OpenMRS were evaluated. Additionally, informal interviews were conducted with organizations conducting clinical trials throughout Kenya.

These discussions and informal interviews brought to the forefront issues related to conducting clinical trials in the same setting that clinical care is provided. These issues were used to create seven requirements and five scenarios of use for an OBDIS. The requirements are:

- R1. The solution must have little or no implementation cost and be maintainable by the local staff.
- R2. The solution must be compatible with currently deployed software.
- R3. The solution must be functional with asynchronous data connectivity.
- R4. The solution must be able to answer questions posed by clinicians and researchers.
- R5. The solution must be flexible enough to incorporate multiple diseases and medical conditions.
- R6. The solution must be usable by non-informaticists or IT personnel.
- R7. The solution must provide information in a timely manner.

In addition to the requirements listed above, Drs. Abernethy, Brinkley, Walson and I created eight scenarios of use. These scenarios are based on first-hand experience: the use of health information systems in low-resource settings (Drs Abernethy, Walson, and myself), and in developing health information systems (Drs. Abernethy, Brinkley, and myself). They represent instances in which the integration between EMR and EDC systems would be useful in a medical setting. Five medical situations are described in the scenarios of use: continuity of care, querying across systems, harmonizing scheduling, preventing duplication of effort, and development of clinical trial cohorts. These scenarios were introduced in (80); however, they have been modified and expanded since publication. The full set of modified and expanded scenarios are listed in Table 6 below. These scenarios will be used to evaluate the functionality of the system in Chapter 7.

Scenario Number	Scenario
S1	Querying other provider's systems to identify prior medical records for a given patient.
S2	Determining the inclusion criteria for clinical trials, and querying systems to determine patients who will be included or excluded from the trial.
S3	Identifying adverse events, routine monitoring, side effects or contraindications caused by routine clinical care or clinical trial study protocols.
S4*	Sharing information for the optimal care of patients is often a challenge if the population is mobile, if treatment is sought opportunistically from among the few available resources, or if patients seek care from different sources for distinct conditions. In the best scenario, a patient might present with a paper record from another clinic summarizing their treatment or vaccination history. The ability to transfer patient data between currently siloed systems could avoid complications such as misdiagnosis or contraindicated therapy. (For example, a paper-based transfer record could omit a patient's allergy to Septrin, resulting in another provider initiating Septrin prophylaxis, precipitating a severe reaction).
S5*	Clinics may merge operations or client bases, or they may upgrade their systems to a new data format. This will require translation of patient information and clinical data between two data models.

**Table 6 - Scenarios of Use (\* Scenarios that are not implemented in the prototype system.)**

## 4.2 Prototype System Architecture

The prototype system was built using the Java programming language on top of the Struts2 Web Application Framework (95). The Struts2 jQuery plugin and the Apache web server running on Mac OS X were also used. The databases to be integrated are accessed through an ODBC (Open Database Connectivity) connection; the prototype uses MySQL and PostgreSQL. Three query languages were

used: SPARQL, Structured Query Language (SQL) and XQuery. SPARQL and XQuery queries were run using the ARQ (96) and Saxon (97) query engines, respectively. All software and libraries used in creating and running the system are freely available with the exception of the Mac OS X operating system; however, this can be replaced by any version of Linux.

The OBDIS system architecture consists of three core components: 1) the user interface (UI), 2) the processing component, and 3) Patient Matching Component. Figure 11, below, provides a visual representation of these components and how they interact with each other.

#### 4.2.1 User Interface Component

The UI, component 1, is the mode through which the user interacts with the system. It provides forms by which the user can input query parameters, and results pages, which display query results in tabular format. Three types of query screens are available in the current version of the system: 1) patient identifier, 2) cohort, and 3) scenario-based queries. The UI will be discussed in detail in the System Functionality section (Section 4.3) of this Chapter.

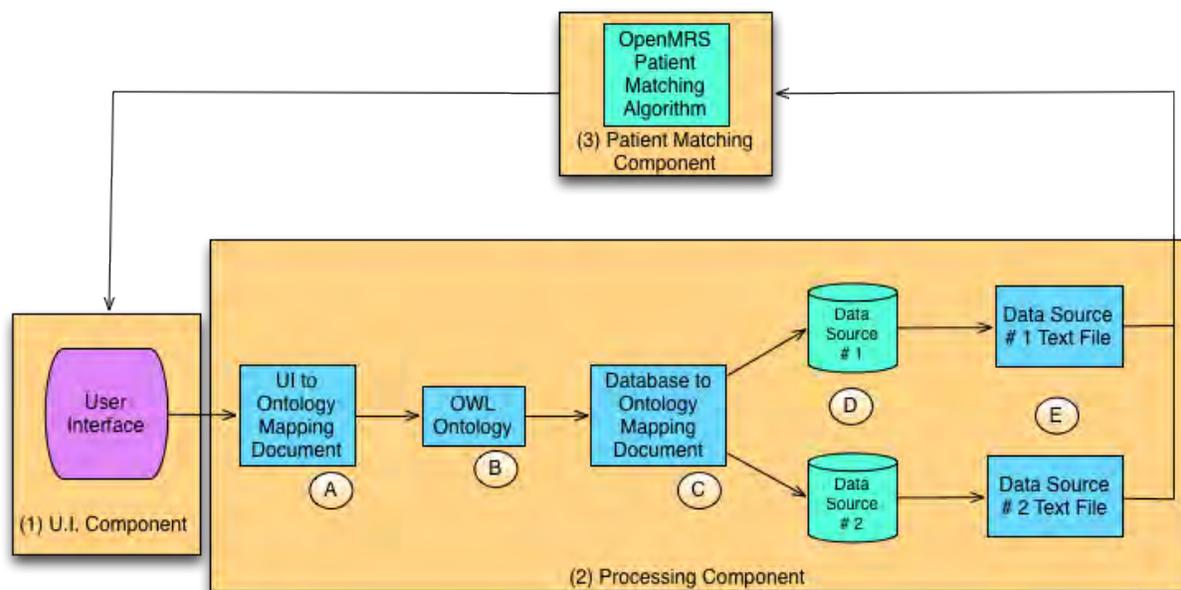


Figure 11 - System Architecture by Component

## 4.2.2 Processing Component

The Processing Component is made up of five parts: A) UI to Ontology Mapping Document (U.I. Mapping Document), B) OWL Ontology, C) Database to Ontology Mapping Document (D.B. Mapping Document), D) Data Sources and E) Data Source Text Files. Each part will be discussed individually. A discussion of how they interact with each other will follow in Section 4.3, System Functionality.

### 4.2.2.1 U.I. to Ontology Mapping Document

The U.I. Mapping Document (A, Figure 11) provides a mapping between the user interface and the OWL ontologies used in the system. Each query screen is represented in this document. Figure 12, below, provides a look at the patient identification query screen's representation in this document.

```
<UIOntMap>
  <qScreen name="ptIDQuery">
    <hasInput name="ptQClinicID" ontology="ptIdentifierOntologyv3.owl" ontConcept="hasClinicID"/>
    <hasInput name="ptQFirstName" ontology="ptIdentifierOntologyv3.owl" ontConcept="hasFirstName"/>
    <hasInput name="ptQMiddleName" ontology="ptIdentifierOntologyv3.owl" ontConcept="hasMiddleName"/>
    <hasInput name="ptQLastName" ontology="ptIdentifierOntologyv3.owl" ontConcept="hasLastName"/>
    <hasInput name="ptQAge" ontology="ptIdentifierOntologyv3.owl" ontConcept="hasAge"/>
    <hasInput name="ptQDateOfBirth" ontology="ptIdentifierOntologyv3.owl" ontConcept="hasDOB"/>
    <hasInput name="ptQGender" ontology="ptIdentifierOntologyv3.owl" ontConcept="Gender"/>
  </qScreen>
</UIOntMap>
```

**Figure 12 - U.I. to Ontology Mapping Document; Patient ID Query Screen**

The creation of this document is not automated and must be edited by the implementer if changes to the U.I. or the ontologies occur. *UIOntMap* is the root element of the XML document. *qScreen* and *has-Input* are child and grandchild elements of the *UIOntMap* element, respectively. The name of each query screen is the value of the *name* attribute of the *qScreen* element. *has-Input* is the child element of *qScreen*, and has the attributes *name*, *ontology*, and *ontConcept*. The value of the *name* attribute is the field name from the User Interface (e.g. *ptQAge*, *ptQDateOfBirth*). Attribute *ontology*'s value is the file name of the ontology that matches the *ontConcept*'s value. The final attribute *ontConcept*'s value is the ontology concept that matches the field name.

There are no optional attributes for this document. If two ontology concepts are required to correctly map the U.I. field to the ontology, a comma should separate them.

### 4.2.2.2 OWL Ontology

Two ontologies (B, Figure 11), the PIDO and HIVO described in Chapter 3, are used in the prototype system. These ontologies, together with the XML Mapping Documents (A, B, and C in Figure 11), provide the backbone of the system. These ontologies can be replaced by other OWL ontologies when integrating other data sources. However, changes to the U.I., as well as the U.I. and Ontology to DB mapping documents will need to be made.

### 4.2.2.3 Ontology to Database Mapping Document

A version of C, Figure 11, the D.B. mapping document, must exist for every data source. The format for this document was adapted from (98,99). The mapping document contains information, which maps the OWL ontology to the data source the D.B. mapping document describes. Figure 13 below shows an excerpt from the OpenMRS mapping document for the Person table. Only relevant, non-audit related columns (e.g. date\_created), are represented in this document.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ontologyDBMap>
  <database name="OpenMRS">
    <ontology name="ptIdentifierOntologyv3.owl">
      <map TableCol="person.gender" conceptNumber="" ontConcept="Gender"/>
      <map TableCol="person.gender" conceptNumber="" ontConcept="Female"/>
      <map TableCol="person.person_id" conceptNumber="" ontConcept="hasClinicID"/>
      <map TableCol="person.person_id" conceptNumber="" ontConcept="NumericID"/>
      <map TableCol="person.birthdate, person.birthdate_estimated" conceptNumber="" ontConcept="hasDOB"/>
      <map TableCol="person.gender" conceptNumber="" ontConcept="Male"/>
      <map TableCol="person.gender" conceptNumber="" ontConcept="hasGender"/>
      <map TableCol="person.person_id" conceptNumber="" ontConcept="Identifier"/>
    </ontology>
  </database>
</ontologyDBMap>
```

Figure 13 - Database to Ontology Mapping Document; OpenMRS Person Table

A PHP script was created to automate the process of building the basic structure of the document. The script takes as input the database and ontology names, and the OWL ontology file. Output of the script is a partially completed D.B. mapping document, which includes the ontology concepts. The *TableCol* and *conceptNumber* attributes of the *map* element are created by the script, but are left blank. The implementer must complete these attributes.

*ontologyDBMap* is the root element of the XML document, its child, grandchild and great-

grandchild elements are: *database*, *ontology* and *map*, respectively. *database* has attribute *name*, whose value is the database name. This attribute should appear only once per mapping document. *ontology* has attribute *name*, whose value is the ontology name. This attribute should appear once for each datasource. *map* has two required attributes and one optional attribute. The required attributes are: *TableCol*, and *ontConcept*. The optional attribute for the *map* element is *conceptNumber*. The *TableCol* attribute's value is the table name and column name associated with the database table and column concatenated by the '.' operator. The *ontConcept* attribute's value is the ontology class or property, which relates to the database table and column described in the *TableCol* attribute.

The optional attribute of the *map* element is *conceptNumber*. The *conceptNumber* attribute is useful for object relational databases, (e.g. OpenMRS and OpenClinica) and should contain the number value of the concept. In the instance of OpenMRS, a concept's concept ID would be the value of the concept number attribute. (E.g. Pneumonia's concept ID could be 568.) Any queries pertaining to pneumonia would need to contain the concept ID, 568, to correctly identify Pneumonia. (This will be discussed further in Section 4.3, System Functionality, of this Chapter.) If the database were OpenClinica, the clinical study ID would be the value of a concept number attribute. Any combination of the required and optional attributes is acceptable; however, each attribute should only appear once per *map* element. A comma should separate multiple values for each attribute.

#### **4.2.2.4 Data Sources**

The current system only accepts data sources (See D, Figure 11) in database format. The prototype system uses MySQL and PostgreSQL, to access OpenMRS and OpenClinica, respectively. Both systems are implemented using an object relational database; however, relational databases can be used in the prototype system as well. Other open source and proprietary databases with an ODBC connection can be used with the prototype.

#### **4.2.2.5 Data Source Text Files**

These files (See E, Figure 11) are generated by the system after a query has been executed over a data source and before the Patient Matching Component has been initiated. One file exists per data source queried and a new file is created for each executed query. These files are in CSV format and

contain all patient identifying information available from the data source. They are used as input to the OpenMRS Patient Matching Module.

#### 4.2.3 Patient Matching Component

This component is based on the OpenMRS Patient Matching Module (100). The module was created by Shaun Grannis, MD and James Egg, though contributions have been made by other members of the OpenMRS community. In an effort to reduce the runtime and complexity of the module, the functionality has been duplicated. The Levenshtein (101), longest common substring (LCS) (60) and Jaro-Winkler (102) similarity metric algorithms are implemented. The module has many optional features. However, for the purposes of this dissertation, only the Levenshtein algorithm, along with blocking based on gender, was used. The Levenshtein edit distance calculates the number of additions, deletions and replacements required to transform String 1 to String 2 (101).

Blocking reduces the number of comparisons the algorithm must perform by separating the list of patients to be compared based on a particular data column, gender for the purposes of this research. If both male and female patients are returned from the data source queries, male patients will only be compared with other male patients to identify matches, likewise with female patients.

The Patient Matching Module uses a combination of the Expectation Maximization (EM) (81) and Fellegi-Sunter (104) algorithms to determine matches within and between data sources. In addition to those algorithms, random analyzers are used to provide the initial low threshold weight needed for the EM algorithm.<sup>3</sup> Open source Java versions of these algorithms were obtained from (105) and (106).

The system compares the Data Source Text Files against each other. Given two Data Source Text Files, A and B, each patient (row) in file A is compared pairwise to each patient (row) in file B.

The matching module returns a “|” delimited text file. This new file contains the results from both Data Source Text Files. Each row contains two patient rows (the two patients who were compared) and one initial column (see Figure 14 below). The first row of Figure 14 has been added for clarity. The initial column of text in each row is a number, a probability. This number indicates the probability that these two patients are the same person. The OpenMRS Module represents matches using positive and negative

---

<sup>3</sup> As it is not within the scope of this research, I will not describe in detail the Levenshtein, LCS, Jaro-Winkler, EM or Fellegi-Sunter algorithms.

numbers. To make the system more user friendly, probabilities are now represented using decimal numbers ranging from 0 – 1. Numbers closer to 0 (0.10, Figure 14) indicate that the patients are not considered to be a match, “No Match.” Alternatively, numbers closer to 1 (0.95, Figure 14) indicate that the patients are a “Match.” This resulting text file is transformed and output to the user in HTML.

Probability	Pt1Num	Pt2Num	Pt1First	Pt2First	Pt1Middle	Pt2Middle	Pt1Last	Pt2Last	Pt1Age	Pt2Age	Pt1Gender	Pt2Gender
0.45	10	3	Madsion	Mary	Jeanette	Joyce	Bergerton	Martin	5	76	f	f
0.65	10	9	Madsion	Madison	Jeanette	Jeanette	Bergerton	Bergeron	5	5	f	f
0.45	3	19	Mary	Madison	Joyce	Jeanette	Martin	Bergeron	76	5	f	f
0.10	2	4	James	Nick	Joyce	J	Batiste	Cooper	31	54	m	m
0.45	2	15	James	N	Joyce	James	Batiste	Cooper	31	54	m	m
0.45	2	6	James	Chuck	Joyce	Daniel	Batiste	Morris	31	28	m	m
0.45	2	7	James	Charles	Joyce	Daniel	Batiste	Morris	31	27	m	m
0.35	2	8	James	James	Joyce	Lucas	Batiste	Scott	31	15	m	m
0.95	4	5	Nick	N	J	James	Cooper	Cooper	54	54	m	m
0.4	4	6	Nick	Chuck	J	Daniel	Cooper	Morris	54	28	m	m
0.45	4	7	Nick	Charles	J	Daniel	Cooper	Morris	54	27	m	m
0.45	4	8	Nick	James	J	Lucas	Cooper	Scott	54	15	m	m
0.45	5	6	N	Chuck	James	Daniel	Cooper	Morris	54	28	m	m
0.45	5	7	N	Charles	James	Daniel	Cooper	Morris	54	27	m	m
0.45	5	8	N	James	James	Lucas	Cooper	Scott	54	15	m	m
0.65	6	7	Chuck	Charles	Daniel	Daniel	Morris	Morris	28	27	m	m
0.45	6	8	Chuck	James	Daniel	Lucas	Morris	Scott	28	15	m	m
0.45	7	8	Charles	James	Daniel	Lucas	Morris	Scott	27	15	m	m

**Figure 14 - Patient Matching Algorithm Result; Initial row added for clarity**

### 4.3 System Functionality

After logging into the system, the user is presented with three query options: Patient Identification, Cohort and Scenario-based Queries. The Patient Identification Query or “Patient ID Query” is used when searching for a specific patient, Figure 15 below, using a small subset of the identifying information represented in the PIDO. The Cohort Query screen, Figure 16, allows the end user to identify a group of patients that meet certain criteria. Demographic and encounter-level parameters are available to the user. Figure 17 displays the Scenario-Based Queries menu screen. The user is presented with English translations of each scenario and a link. The link takes the user to an abbreviated or concatenated version of the patient identifier and cohort query screens tailored to the particular query.

[Pt ID Query](#) [Cohort Query](#) [Scenario-Based Queries](#) [About](#) [Logout](#)

## Patient ID Query

(Query 1: Find all encounters for Patient X.)

Clinic ID:	<input type="text"/>	
First Name:	<input type="text"/>	
Middle Name:	<input type="text"/>	
Last Name:	<input type="text"/>	
Age*:	<input type="text"/>	1-100
Date of Birth:	<input type="text"/>  (dd-mm-yyyy)	If month and/or day unknown please enter "01".
Gender:	<input type="radio"/> Male <input type="radio"/> Female <input checked="" type="radio"/> Male & Female	
<input type="button" value="Reset"/> <input type="button" value="Submit Query"/>		

\*Age is calculated based on current date.

Figure 15 - Individual Patient Identification Query Screen

## Cohort Query

**Scenario 2: Determining the inclusion criteria for clinical trials, and querying systems to determine patients who will be included or excluded from the trial.)**

Patient Attributes	
Gender:	<input type="radio"/> Male <input type="radio"/> Female <input checked="" type="radio"/> Male & Female
Age*:	between 20 <input type="text"/> and 30 <input type="text"/> years
Birthdate:	between <input type="text"/> <input type="text"/> and <input type="text"/> <input type="text"/>
Patient Status:	<input checked="" type="radio"/> Alive Only <input type="radio"/> Deceased Only <input type="radio"/> Alive or Deceased
Encounter Attributes	
Diagnosis:	equal <input type="text"/> None <input type="text"/> date range on/before <input type="text"/> <input type="text"/> on/after <input type="text"/> <input type="text"/>
	equal <input type="text"/> None <input type="text"/> date range on/before <input type="text"/> <input type="text"/> on/after <input type="text"/> <input type="text"/>
	equal <input type="text"/> None <input type="text"/> date range on/before <input type="text"/> <input type="text"/> on/after <input type="text"/> <input type="text"/>
	equal <input type="text"/> None <input type="text"/> date range on/before <input type="text"/> <input type="text"/> on/after <input type="text"/> <input type="text"/>
Drugs:	equal <input type="text"/> None <input type="text"/> date range on/before <input type="text"/> <input type="text"/> on/after <input type="text"/> <input type="text"/>
	equal <input type="text"/> None <input type="text"/> date range on/before <input type="text"/> <input type="text"/> on/after <input type="text"/> <input type="text"/>
	equal <input type="text"/> None <input type="text"/> date range on/before <input type="text"/> <input type="text"/> on/after <input type="text"/> <input type="text"/>
	equal <input type="text"/> None <input type="text"/> date range on/before <input type="text"/> <input type="text"/> on/after <input type="text"/> <input type="text"/>
Observations:	None <input type="text"/> not <input type="text"/> date range on/before <input type="text"/> <input type="text"/> on/after <input type="text"/> <input type="text"/>
	None <input type="text"/> not <input type="text"/> date range on/before <input type="text"/> <input type="text"/> on/after <input type="text"/> <input type="text"/>
	None <input type="text"/> not <input type="text"/> date range on/before <input type="text"/> <input type="text"/> on/after <input type="text"/> <input type="text"/>
	None <input type="text"/> not <input type="text"/> date range on/before <input type="text"/> <input type="text"/> on/after <input type="text"/> <input type="text"/>
WHO Stage:	equal <input type="text"/> Adult Stage 2 <input type="text"/>
Visit Date:	date range on/before 01-01-1996 <input type="text"/> on/after 31-12-2012 <input type="text"/>
Physician's Name:	equal <input type="text"/>
Clinic Name:	equal <input type="text"/>
<input type="button" value="Reset"/> <input type="button" value="Submit Query"/>	

\*Age is calculated based on current date.

**Figure 16 - Cohort Query Screen**

[Pt ID Query](#) [Cohort Query](#) [Scenario-Based Queries](#) [About](#) [Logout](#)

## Scenario-Based Queries

**Scenario 1: Querying other provider's systems to identify prior medical records for a given patient.**

- [Query 1](#): Find all encounters for Patient X. (Patient ID Query)
- [Query 2](#): Find all future visit dates for Patient X.
- [Query 3](#): Identify recent laboratory data for Patient X.

**Scenario 2: Determining the inclusion criteria for clinical trials, and querying systems to determine patients who will be included or excluded from the trial (Cohort Query)**

**Scenario 3: Identifying adverse events, routine monitoring, side effects or contraindications caused by routine clinical care or clinical trial study protocols.**

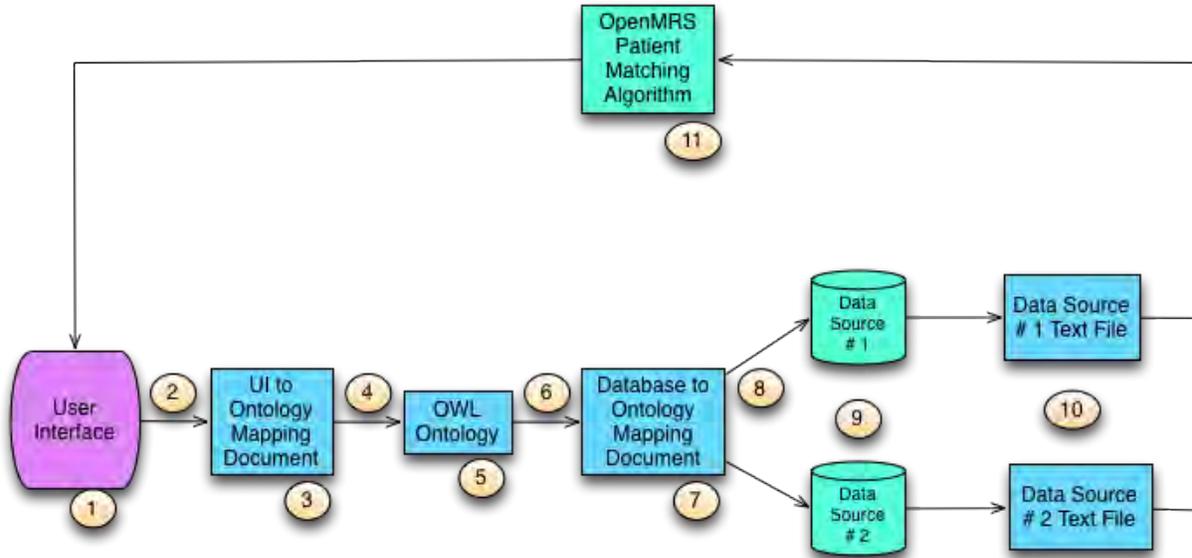
- [Query 1](#): Find all patients of Physician Y who are on drug(s) A and have not had a clinical encounter in the past X months.
- [Query 2](#): Find all patients of Clinic Z, on drug regimen ABC and DEF during time period G.

\*Age is calculated based on current date.

**Figure 17 - Scenario-Based Queries Menu Screen**

I will use Query #1, "*Find all encounters for Patient X,*" along with the System Architecture Workflow in Figure 18 to describe the system's functionality.

Step 1, Figure 18 requires the end user to enter as much information as is available into the query screen of their choice. For Query #1, the Patient ID Query screen (Figure 15 above) is used. Because the system is intended for use in low-resource settings, certain accommodations in regards to age, date of birth, and date have been implemented.



**Figure 18 - System Architecture Workflow**

If Age is used as a query parameter, the patient's date of birth from the data source, along with the current date, is used to calculate age. Date of birth and all dates should be input and are displayed in (DD-MM-YYYY) format. If both date of birth and age are entered as query parameters, date of birth is used instead of age. If the month or day is unknown for any date parameter, the user is instructed to enter "01."

In Step 2, the work of the Processing Component discussed earlier in this chapter (Section 4.2.2) begins. After the user submits their query parameters, each parameter is used to query the U.I. Mapping Document (3, Figure 18), using XQuery and the Saxon query engine to identify the related ontology concept. This result is returned as a text-based list of U.I. field names and ontology concepts concatenated by the "-" operator, Figure 19 below, and is stored in memory for later use.

```

ptQDateOfBirth-hasDOB
ptQGender-Gender
ptQClinicID-hasClinicID
ptQFirstName-hasFirstName
ptQLastName-hasLastName
  
```

**Figure 19 - U.I. to Ontology Mapping XQuery Result. U.I. field name and Ontology Concept separated by a "-".**

Next, a SPARQL query is run over the OWL Ontology (5, Figure 18), PIDO in this case, to return all classes and properties of the ontology using the ARQ query engine. The result of this query is another text-based list of ontology concepts, Figure 20 below.

```

ConceptualThing
Gender
City
isMaritalStatusOf
hasTreatmentSupporter
hasLastName
Single
Person
Investigator
MaritalStatus
Jurisdiction
hasMiddleName
Race
isAttributeOf
isStudyParticipantOf
Female
Village
StudyParticipant
District
Attribute
isLocationOf
isSubLocationOf
hasStudyID

```

**Figure 20 - PtID Ontology SPARQL Query result excerpt.**

The lists of ontology concepts from the OWL Ontology and the U.I. Mapping Document are used as input for a partially dynamically generated XQuery. The structure of this query remains static with the mapping document name and ontology concepts programmatically inserted by the system, based on the system configuration file and the results of the SPARQL query over the OWL Ontology. The query is run over the DB Mapping Document (Step 7, Figure 18), using the Saxon query engine. This query results in a list similar to Figure 19. The ontology concept, and database table and column names are listed and concatenated by a "-". After this step, the list from Figure 20 is no longer necessary.

At this point (Step 8, Figure 18), one text-based list, the combined results of the U.I. to DB mapping query, results of the PIDO DB mapping query, and user query parameters exist. An example of this list can be seen in Figure 21 below.

```

ptQDateOfBirth-person.birthdate
ptQhasDateOfBirth-person.birthdate_estimated
ptQGender-person.gender
ptQGender-person.gender
ptQGender-person.gender
ptQFirstName-person_name.given_name
ptQLastName-person_name.family_name

```

**Figure 21 – U.I. to DB Mapping Document Query result excerpt**

Step 8 is the most important step as it dynamically creates the SQL statements that are executed over the associated data source (Step 9, Figure 18). A SQL query consists of three clauses: SELECT, FROM and WHERE. Each clause is built independently and simultaneously, and later concatenated to create the final SQL query.

The final text-based list is iterated over and parsed. Each line of the list is parsed for three instances:

1. The presence of “ptQ”(patient identifier query), “cq” (cohort query) and “sb” (scenario based query) in the first two characters in the entry identifies the entry as a user input entry and will become a part of the WHERE clause.
2. The presence of “-“ operator in the entry signifies that all characters after this operator should be used for the SELECT clause.
3. The presence of the “.” operator in the entry becomes a part of the SELECT and the FROM clause. Any text before the “.” is a database table name, and anything after is a column name.

When the “-“ is encountered, all characters after this operator are added to the SELECT clause without manipulation and are subsequently parsed for the “.” operator. Once that operator is encountered, all text before the operator is added to the FROM clause.

The SELECT and FROM clauses are straightforward in that a comma separates the entries; however, the WHERE clause handles both the user input and JOINS. When a user input entry is identified, the entry is compared to the list of user input parameters from Step 1, Figure 18. If a match is found, and no *conceptNumber* is required, the user’s input is concatenated with the associated database

table and column names by an “=” sign. If a *conceptNumber* is required, the associated table and column name are concatenated with an “=” and the concept number. If the user provides multiple entries for the same parameter, e.g. diagnosis, drug or observations from the Cohort Query Figure 16, a SQL “IN” clause is used. The diagnosis concept numbers are surrounded in parentheses and separated by commas. If only Patient Identifier information is being queried JOINS are created by concatenating person ID with an “=” along with each table in the FROM list. If medical information is being queried, patient tables are joined with medical data tables using patient IDs, and encounter IDs or their equivalent. Finally, all text-based user input, e.g. name, gender, etc, are added to the WHERE clause using the SQL “LIKE” and “%” operators. If duplicates are found, such as “ptQGender” in Figure 21, the entries are compared and only unique instances are used in the final query.

After the list has been exhausted, the SELECT, FROM and WHERE clauses are concatenated in that order and the query is complete. The process for creating SQL queries is repeated for each data source.

The final SQL queries are run over the appropriate data sources in Step 9 (Figure 18). The results of those queries are transformed into CSV formatted text files (Step 10, Figure 18). These files contain all available identifying patient information, which will be used as input to the OpenMRS Patient Matching Module (Step 11, Figure 18). After processing, this module returns the text file discussed in Section 2.3, Figure 14, of this Chapter. This result is output to the user, via the User Interface (Step 1, Figure 18), in HTML format on the Patient Data Query Results page, Figure 22 below.

[Pt.ID Query Cohort Query Scenario-Based Queries About Logout](#)

**Patient Data Query Results**

Query Parameters: First Name: Jasper Last Name: Ngetch Gender: Male [Edit Query](#)

Clinic ID	Name Prefix	First Name	Middle Name	Last Name Prefix	Last Name	Last Name 2	Last Name Suffix	Gender	Date of Birth	Age*	Marital Status	Clinic ID Matches	Probability	Show Comparison
<a href="#">56364</a>		Jasper			Ngetch			M	05-06-1926	86				
<a href="#">91271</a>		Jasper			Ngetch			M	05-06-1926	86		171	0.47	<a href="#">171</a>
<a href="#">2</a>								m	05-06-1926	86				
<a href="#">3</a>								m	23-04-1968	44				
<a href="#">4</a>								m	23-08-1952	60				
<a href="#">5</a>								m	26-02-1952	61				
<a href="#">6</a>								m	21-01-1981	32				
<a href="#">7</a>								m	28-06-1958	54				
<a href="#">8</a>								m	02-09-1969	43				
<a href="#">9</a>								m	07-03-1919	94				

**Figure 22 - Patient Data Query Results Screen**

This page provides the user with a list of basic demographic information about the patients that

meet the user's query parameter(s). The "ClinicID Matches" column displays provide a comma-delimited hyperlinked list of patient clinicIDs of the patients with a match probability of 0.45, or higher. It should be noted that all patients returned meet the user's query parameters. The probabilities only provide information as to the probability to which the patients could be the same person. The probability column provides a comma-delimited list of all the probabilities that correspond to the clinicIDs in the "Clinic ID Matches" column. Show Comparisons provides a hyperlinked list of clinicID's, which transports the user to the Matching Comparison Results screen.

This screen allows the user to see all patients in the ClinicID Matches column in tabular format for comparison. Hyperlinked ClinicIDs allow the user to view the individual patient's Encounter History. To the right of the table are checkboxes, which allow the user to link the checked patients and/or link all patients as necessary. This allows the user to see all checked patients as one patient, instead of two.

[Pt ID Query](#) [Cohort Query](#) [Scenario-Based Queries](#) [About](#) [Logout](#)

### Matching Comparison Results

Clinic ID	<a href="#">171</a>	<a href="#">91271</a>
Probability		
Prefix		
First Name		Jasper
Middle Name		
Last Name Prefix		
Last Name		Ngetch
Last Name 2		
Last Name Suffix		
Gender	m	M
Date of Birth	05-04-1966	05-06-1926
Age	47	86
Marital Status		

171  91271  
 [Link All Patients](#)

\*Age is calculated based on current date.

**Figure 23 - Matching Comparison Results Screen**

Clicking on an ID in the Clinic ID or Clinic ID Matches columns in Figure 22 or the ClinicID in Figure 23, opens the Patient Encounter History Screen, Figure 24. This page provides the user with

basic demographic information on the selected patient: Patient ID, Name, Gender, DOB and Age (calculated based on current date). Additionally, a list of all patient encounters is listed in descending date order. If the information is available, the encounter type (Initial or Return), visit type (Scheduled or Unscheduled), and the clinic name where the encounter occurred are also displayed. Finally, navigational links are available that allow the user to click on an encounter date or view the patient's vital signs and lab results history over all patient encounters.

The date link opens up the Patient Encounter Screen, shown in Figure 25 below. This screen provides: the patient's basic demographic information; basic encounter information; the patient's vital signs; all information/observations obtained and finally, a list of all lab results for the encounter.

The final navigation link on the Patient Encounter History Screen takes the user to the Patient Vital Signs and Lab Results History, Figure 26 below. Again, the basic patient demographic information is displayed along with a list of all vital signs and lab results by encounter.

If a Clinic ID (Figure 22) or Date (Figure 24) is clicked, the queries executed by the system are different than those executed when the Pt ID Query screen is submitted. Instead of querying the Patient Identifier Ontology (Step 5, Figure 18), the system uses the HIV Ontology and bypasses Steps 10 & 11, Figure 18. These steps are no longer required because patient matching is not necessary -- the patient in question has already been identified.

A walkthrough of the system using screenshots in order by Scenario can be found in Appendix H.

[Pt ID Query Cohort Query Scenario-Based Queries About Logout](#)

Patient ID: 56364-openMRS

Patient: Ngetch , Jasper

Gender:M DOB: 05-06-1926 Age: 86 [View All Patient Vitals and Test Results](#)

Date	System	Encounter Type	Visit Type	Clinic Name
<a href="#">2012-10-29</a>	openMRS	Adult Return		
<a href="#">1999-01-25</a>	openMRS	Adult Initial		
<a href="#">1990-12-19</a>	openMRS	Adult Return		
<a href="#">1974-10-17</a>	openMRS	Adult Return		

\*Age is calculated based on current date.

Figure 24 - Patient Encounter History Screen

[Pt ID Query Cohort Query Scenario-Based Queries About Logout](#)

Patient ID: 2326  
 Patient: Ngetch , Jasper  
 Gender: M DOB: 05-06-1926 Age: 86  
 Date: 10/12/28 Visit Type: Adult Return Scheduled/Unscheduled:

**Vital Signs**

VISIT TYPE	Unscheduled visit
FUNCTIONAL STATUS	Functional Status
FUNCTIONAL STATUS	WORKING
CURRENT WHO HIV STAGE	WHO STAGE 3 ADULT
CURRENT WHO HIV STAGE	WHO class 3 adult
CURRENT WHO HIV STAGE	WHO class III adult
POTENTIAL MEDICATION SIDE-EFFECTS OR OTHER PROBLEMS	ABDOMINAL PAIN
SEVERITY OF SIDE-EFFECTS	MODERATE
NEW SYMPTOMS/DIAGNOSES/OPPORTUNISTIC INFECTIONS	Herpes Zoster
REASON FOR DISCONTINUATION OF PROPHYLAXIS MEDICATION	CONDOMS
ADHERENCE TO PROPHYLAXIS MEDICATION	GOOD ADHERENCE
ADHERENCE COTRIMOXAZOLE	FAIR ADHERENCE
ARV ADHERENCE ASSESSMENT	DIAPHRAGM
ARV ADHERENCE ASSESSMENT	CERVICAL CAP
ARV ADHERENCE ASSESSMENT	Cervical contraceptive cap
Reason for poor treatment adherence	CANNOT AFFORD TREATMENT
Reason for missing medication dose	CANNOT AFFORD TREATMENT
Reason for poor medication compliance	CANNOT AFFORD TREATMENT
TESTS ORDERED	CD4 PANEL
TESTS ORDERED	LYMPHOCYTE SUBSET PANEL
ANTIRETROVIRAL DRUG NAME	LOPINAVIR / RITONAVIR
ANTIRETROVIRAL DRUG NAME	Kaletra
PROPHYLAXIS MEDICATION NAME	DAPSONE
PROPHYLAXIS MEDICATION NAME	Aczone

**Lab Results**

Figure 25 - Specific Patient Encounter Screen

[Pt ID Query](#) [Cohort Query](#) [Scenario-Based Queries](#) [About Logout](#)

**Patient ID:** 2312

**Patient:** Sing'Ogei , Callistus Chepkirui

**Gender:** M **DOB:** 1982-02-18 **Age:** 30

Vital Signs/Lab Results	
<b>04-04-2006</b>	
Vital Signs	
DIASTOLIC BLOOD PRESSURE	60
PULSE	80
SYSTOLIC BLOOD PRESSURE	100
TEMPERATURE (C)	36
Lab Results	
BLOOD OXYGEN SATURATION	98
WEIGHT (KG)	58
CD4 COUNT	142
CD4%	14
CD8 COUNT	732
<b>02-05-2006</b>	
Vital Signs	
DIASTOLIC BLOOD PRESSURE	60
PULSE	95
SYSTOLIC BLOOD PRESSURE	100
TEMPERATURE (C)	37
Lab Results	
BLOOD OXYGEN SATURATION	96
WEIGHT (KG)	55

Figure 26 - Patient Vital Signs and Lab Results History

## 4.4 Application Deployment

One of the benefits of my approach is the ability to deploy the application with minimal knowledge of software development and installation, and server administration. If the current user interface and ontologies are used, the implementer need only edit the system's configuration file and create mapping documents for each data source. The configuration file (XML format) provides the system with information as to the location and login information of databases, and the location of mapping documents. After editing and/or creating these documents, the implementer should deploy the war file and restart the Tomcat server. See Appendix I for detailed installation instructions.

## 4.5 Conclusions

I have described the requirements, which were used to develop the OBDIS. Next, I detailed the system's architecture and how the system works with the use of a user scenario. Finally, I described the steps an implementer needs to take in order to implement the system. The prototype provides a homogeneous view of data from integrated sources and integrates OpenClinica and OpenMRS, EDC and EMR systems, respectively. Individual patient and patient cohort queries are also available to the user. Additionally, configuration files and dynamically created queries reduce the burden on implementers when deploying the system. The next chapter, Chapter 5, will describe the data used to evaluate the patient matching algorithm and the prototype system. Following that, Chapters 6 and 7 will evaluate the patient matching algorithm and the prototype system itself.

## Chapter 5: Simulated and Real-World Data

Two types of data – simulated and real-world were used to evaluate the patient matching algorithm and the prototype system. In this chapter, I describe the contents of the simulated dataset and its creation, as well as the data sources that make up the real-world dataset.

### 5.1 Simulated Data Creation

#### 5.1.1 Overview

Two PHP scripts were developed to create initial and follow-up visit simulated datasets to represent clinical research and clinical visit data. These datasets were then used as input for PHP scripts that load the data into the associated databases, OpenMRS and OpenClinica. While the initial visit dataset includes demographic data, it is described separately as it was used to evaluate the patient matching algorithm.

#### 5.1.2 Demographic Data

In order to create realistic demographic datasets that reflect the types of data found in Kenya, four external datasets were used: male first names, female first names, Kenyan surnames and Kenyan health facility locations. All external datasets were obtained in CSV format, if available. If not, they were transformed into CSV format. Table 7, below, provides a detailed list of the datasets, their locations, a brief description and the variables for which the dataset was used in creating the simulated datasets.

<b>Dataset Name</b>	<b>Location</b>	<b>Description</b>	<b>Variables</b>
Male Names	<a href="http://www.babycentre.co.uk/pregnancy/naming/baby-names-2011/babycentre-top-boys-names-2011/">http://www.babycentre.co.uk/pregnancy/naming/baby-names-2011/babycentre-top-boys-names-2011/</a>	Top 100 UK boy's names 2011	Male First Name
Female Names	<a href="http://www.babycentre.co.uk/pregnancy/naming/baby-names-2011/babycentre-top-girls-names-2011/">http://www.babycentre.co.uk/pregnancy/naming/baby-names-2011/babycentre-top-girls-names-2011/</a>	Top 100 UK girl's names 2011	Female First Name
Kenyan Surnames	OpenMRS Sample database	List of Kenyan surnames (2,271 names)	Family Name
Kenyan Health Facility	<a href="http://www.ehealth.or.ke/facilities/downloads.aspx">http://www.ehealth.or.ke/facilities/downloads.aspx</a> Obtained September 8, 2012	Detailed list of health facilities, their locations, services and contact information (8,321 facilities) <sup>4</sup>	Address, Facility, District

**Table 7 - Simulated Data External Datasets**

The male and female first name datasets were obtained from the Babycentre Baby Names website (107,108). Both the male and female lists have one hundred names. These datasets were chosen because Kenyans often use Christian names as their first name when conducting business (Personal Communication Edwin Wambua, October 2012). The Kenyan surnames were obtained from

<sup>4</sup> A sub-list of the first one hundred was used.

the OpenMRS sample database (109). A SQL query was run over the sample database and the results were exported in CSV format. This yielded a total of 2,271 names. Finally, the Kenyan health facility dataset was obtained in CSV format from the Kenyan e-health website (110). A total of 8,321 facilities were included in the original dataset; however, only the first one hundred facilities were used in creating the simulated data. Only health facilities registered with this site are included. It is possible that other facilities exist. The district and location information from this dataset were used to simulate data for district and address.

Ages 18 – 100 were used, specifically the 1912-01-01 to 1994-12-31 date range. The unique ID and clinic ID variables were randomly selected from the following bins of numbers, 77000-99999 and 44000-69999, respectively. Address and telephone number were formatted based on the Kenyan standard and again randomly generated with the associated variables seeded with data from the health facilities' CSV file.

### **5.1.3 Clinical Visit Data**

The clinical visit dataset was created using the WHO HIV/AIDS Minimum dataset guidelines discussed in Chapter 3. Two datasets were created -- adult initial visit and adult follow-up visit. The adult initial visit dataset consisted of 110 variables and the adult follow-up visit dataset had 42 variables.

Where available, the coding suggested by the WHO guidelines was used. These coded variables were chosen randomly using the rand() PHP function.<sup>5</sup> Dates were randomly generated based on the patient's age.

Non-demographic variables were randomly generated based on the previously created demographic variables (age at registration of HIV care, date of birth and gender).

The adult initial dataset resulted in 1,000 patients, 110 variables and 110,000 data points. The adult follow-up visit dataset resulted in 42 variables and 104,622 data points.

---

<sup>5</sup> <http://php.net/manual/en/function.rand.php>

#### 5.1.4 Clinical Research Data

The clinical research data set was created based on a simulated clinical research study. The simulated study looks at HIV and AIDS patients between the ages of 18 and 50, approximately 549 patients total.

After the enrollment visit, patients returned for follow-up visits at three-month intervals over a two-year period.

The enrollment dataset resulted in 549 patients, 93 variables and 51,150 data points. The adult follow-up visit dataset resulted in 97 variables and 333,680 data points.

#### 5.1.5 Data Overlap

Of the 549 patients in the Clinical Research Dataset, 149 were patients from the Clinical Visit Dataset. This overlap in patients not only represents the idea that patients enrolled in clinical trials also receive clinical care in local clinics, but that duplicate patient data can be represented in multiple siloed systems. The Venn diagram, Figure 27, below provides a visual example of the overlap.

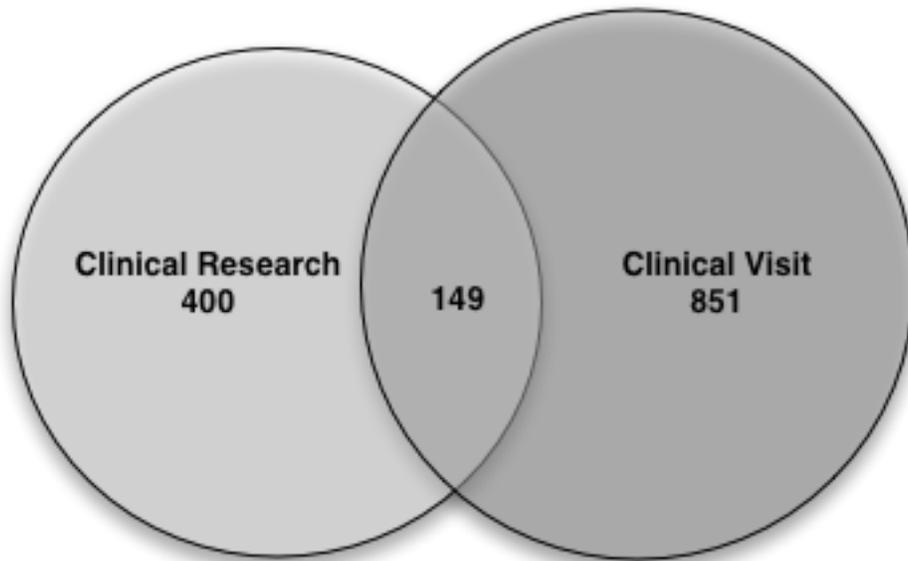


Figure 27 - Simulated Data: Clinical Research versus Clinical Visit Venn diagram

### 5.1.6 Data Import

The CSV formatted simulated datasets were used as input for PHP scripts that transform the datasets into SQL insert statements associated with the concepts (OpenMRS) and forms (OpenMRS and OpenClinica) from each system. Once executed, the scripts programmatically load the data into the appropriate database.

## 5.2 Real-World Data Sources

### 5.2.1 Clinical Visit Data

The clinical visit dataset was obtained from the KEMRI-Wellcome Trust Programme's HIV clinic operated in the Kilifi District Hospital on the eastern coast of Kenya. Five datasets exported from their system were used in this study: Adult intake; Child intake; clinical encounter; anthropometric; and laboratory data. This dataset was de-identified; therefore, full patient names and addresses were omitted. However, patient initials were included and were used instead of full names for the given (first name), middle and family name (last name) data fields. Clinic identification numbers were used to match patients between datasets. Therefore, if a clinic identification number was missing from the dataset, the patient or encounter (clinical visit) was excluded from the data import. Moreover, if incomplete data dictionary entries were supplied, that data was excluded as well.

The Adult intake dataset is made up of demographic data and is only collected once, at enrollment into care. Patients are considered adults if they are age 18 or older. In addition to the standard demographic data (gender, date of birth and marital status), other data related to the patient's sexual history, HIV status, and partner's HIV status are included. This dataset describes 4,207 patients, and has 139,379 observations.

The Child intake dataset is also made up of data collected only at enrollment into care for patients under the age of 18. Like the Adult intake dataset, the child's standard demographic information and HIV status are collected. However, instead of focusing on sexual history, the majority of variables are related to the child and child's parent's HIV status. This dataset describes 1,485 patients, and has 36,679 observations.

Anthropometric data is collected at every client visit. The patient's height (or length), weight, head and mid-upper arm circumference, in addition to data about a pregnant woman and her child's breastfeeding status are recorded. Finally, the patient's visit type (first visit, scheduled, unscheduled), ARV and cotrimoxazole medication usage (to prevent O.I.'s) are collected. This dataset describes 26,592 encounters and has 326,488 observations.

All data collected during encounters with the clinician are included in the clinical encounters dataset. These variables are related to the patient's WHO HIV stage, newly acquired opportunistic infections, current anti-retroviral medication and medication adherence. 44,592 encounters and 2,497,135 observations were included in this dataset.

The final dataset, the laboratory data, are routine tests requested by clinicians. In addition to date of sample collection, twenty laboratory values are collected, all of which are from blood samples. Routine laboratory tests such as blood cell and CD4 counts are collected in addition to other blood-related tests that indicate disease stage. This dataset describes 7,623 encounters and 59,752 observations.

### **5.2.2 Clinical Research Data**

The OpenClinica instance was populated with data from a clinical trial run in Kenya from 2007 - 2011, with Principal Investigator Judd L. Walson, M.D. This study, entitled "Empiric Therapy of Helminth Co-infection to Reduce HIV-1 Disease Progression" (THE or PHE), collects various data points related to the patient's living situation and medical condition (94). This study has enrolled approximately 940 patients, across three sites in Kenya (Kisii, Kisumu, Kilifi) (Personal Communication Linda Chaba, 31.May.2010). The patients come to the clinic for a baseline evaluation and then again every three months, completing eight visits over a period of two years. The CRFs used in this study were created in OpenClinica. The data obtained from Dr. Walson's study was de-identified and does not contain patient names, addresses or study ID numbers. However, for the purposes of establishing a gold standard against which to evaluate the patient matching algorithm, the clinic ID (the identification number provided by the study) was used since this ID directly links individual patients in the clinical and research systems.

### 5.2.3 Data Overlap

The clinical visit and clinical research datasets were chosen because of their overlap. The Kilifi site for the THE study is the same location where the Wellcome Trust clinic is located. Because of this, some of the patients in the THE study are included in the Wellcome Trust dataset, alternatively, there are some patients in the Wellcome Trust dataset that are not participants in the THE study. The Venn diagram in Figure 28, below, provides a visual representation of the overlap in datasets. Specifically, of the 5,692 total patients in the Wellcome Trust dataset only 3,965 were used because of incomplete data. This resulted in a total of 4,716 patients comprising the Real World Dataset, of those patients 3776 are unique to the Wellcome Trust, 751 to the THE study and 189 were included in both datasets.

### 5.2.4 Data Import

All datasets were obtained in CSV format. Like the simulated data import, the datasets were input into a PHP script that transforms the datasets into SQL insert statements associated with the concepts and forms from each system.

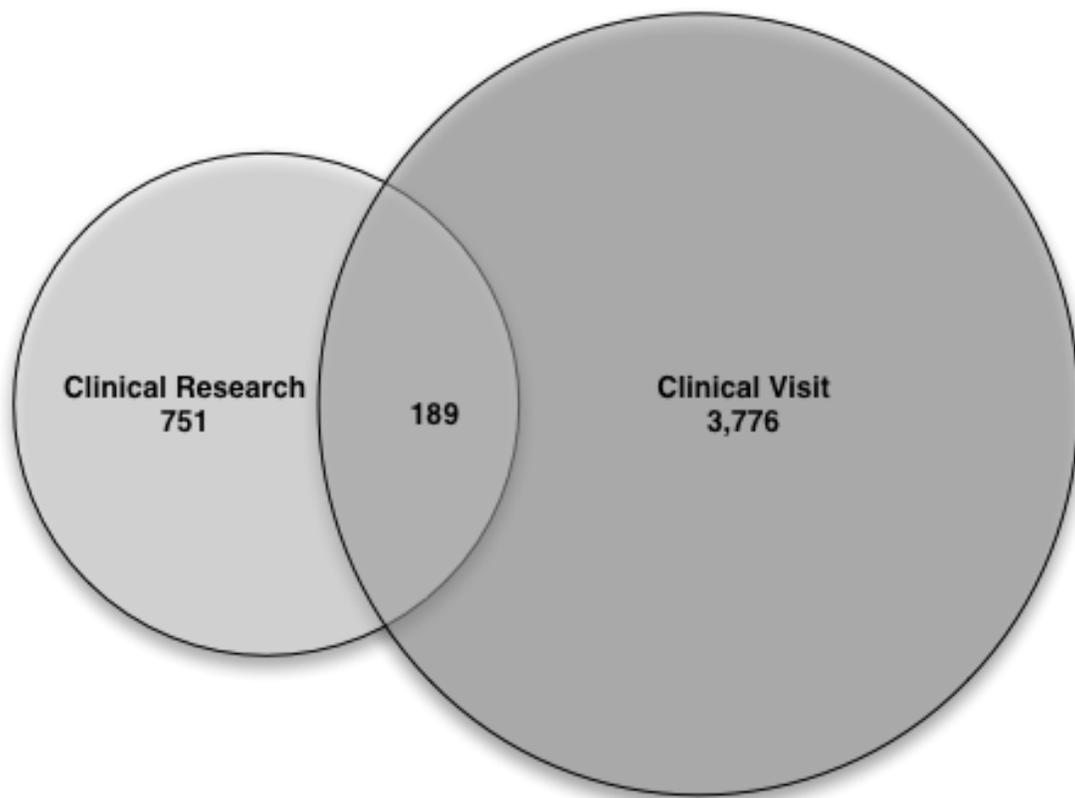


Figure 28 - Real World Data: Clinical Research versus Clinical Visit Venn Diagram

### 5.3 Conclusions

The datasets described in this chapter mimic the types of data created and used in the HIV/AIDS clinical research and clinical visit setting and will be beneficial in the evaluation of the prototype system. The next two chapters, 6 and 7, will describe how these datasets were used to evaluate the patient matching algorithm and the prototype system, respectively. These evaluations were used to determine whether the prototype is successful in correctly identifying patients and providing access to data from disparate sources.

## Chapter 6: Patient Matching Algorithm Evaluation

Searching an EMR for information about a patient participating in a clinical research study with a common clinic ID would not warrant the use of the PIDO or a matching algorithm. Using the clinic ID, the user and the system would be able to correctly identify the associated patient. However, it is unlikely that a patient would have the same clinic or study ID across disparate systems. The patient-matching algorithm, used in the prototype system, utilizes probabilistic matching to identify patients across systems without a common clinic ID. This chapter focuses on the evaluation of the patient-matching algorithm and is followed by Chapter 7, which describes the evaluation of the prototype system as a whole.

The patient-matching algorithm was evaluated in two phases. This two-phase evaluation allowed us to determine the algorithm's performance when matching patients without a common ID across systems under various real world conditions. In Phase 1, using simulated and real world data, I determined whether the algorithm could correctly identify the matching patient given complete data under certain conditions. In Phase 2, using simulated data, I studied the minimum data points necessary to correctly identify a match.

### 6.1 Evaluation Metrics

The results for both phases were reviewed for answers to the following three questions:

1. Is the matching patient present in the results list?
2. Is the matching patient among the top five in the results list?
3. Is the matching patient number one in the results list?

The answers to these questions were used to calculate the number of times the probe patient, the patient being searched for, appears in the list and the rank of results in this list. True positives and true negatives were calculated based on the answers. If the probe patient is a member of the overlap (patient appears in both the clinical research and the clinical visit) and the patient was identified, it is considered to be a true positive. Non-overlapping patients who were correctly not identified were considered to be true negatives.

## 6.2 Phase 1: How sensitive is the algorithm given complete data?

In Phase 1, I determined whether the algorithm could correctly identify a matching patient given all data points except Clinic ID (which was used to create the gold standard). Using simulated and real world data, I matched patients using the 16 data points that the patient matching algorithm employs.<sup>6</sup>

The data points are as follows:

1. Name Prefix
2. First Name
3. Middle Name
4. Last Name Prefix
5. Last Name
6. Last Name 2
7. Last Name Suffix
8. Address
9. City
10. State
11. Country
12. Zip Code
13. Age
14. Gender
15. DOB
16. Marital Status

In addition to being the gold standard, the Clinic ID is not included as input based on the assumption that a patient would not have the same Clinic ID in different databases and its inclusion could give a false idea that the matching algorithm works by increasing the probability of an correct match.

### 6.2.1 Methods

Phase 1 of the evaluation was conducted using all patients from the clinical visit and clinical research datasets. List A (Probe List) was comprised of the clinical research dataset, and List B, the clinical visit dataset. This evaluation was performed using simulated and real world patients.

Each patient in the list of probe patients (List A) was used as input into the patient-matching algorithm along with a list of all patients from the clinical research database (List B). A breakdown of the number of patients in each list can be found in Sections 5.1.5 and 5.2.3 of Chapter 5. One patient was excluded from the simulated List B dataset because of incomplete data.

---

<sup>6</sup> This list can be modified based on available data.

Only the first 20 patients returned by the algorithm were used in this evaluation. The result set was limited to 20 based on the assumption that a user would experience information overload if more than 20 results were returned. Moreover, research has shown that most users do not view more than two pages of search results when 10 results per page are returned (111–113). The output of the algorithm was a list of two clinicIDs, one from List A and one from List B, and their associated match probability. This list was sorted in descending order by probability. When order is important (i.e., first patient in the list or one of the top five patients in the list), all patients with the same probability are considered to be the same rank. Example: If ten patients have a probability of 0.75, and the first patient with this probability is in spot five, then all ten patients are considered to be number five in the list. The total number of true positives and true negatives were recorded for each question.

### 6.2.2 Simulated Data Results

Table 8 below, displays the results of the Phase 1 simulated data evaluation by question. For all questions, the probe patient was always the first person in the list, if a match existed. If a match did not exist, the probe patient did not show up in the list.

	True Positive	True Negative
<b>Question 1:</b> <b>Slots 1 - 20</b>	100	100
<b>Question 2:</b> <b>Slots 1 - 5</b>	100	100
<b>Question 3:</b> <b>Slot 1</b>	100	100

**Table 8 - Phase 1 Simulated Data: Percentage of times patient (%) was found in the list by question**

### 6.2.3 Real World Data Results

Like simulated data, for all questions, the probe patient was number one in the list if a match existed and did not appear in the list if a matching patient did not exist. This is reflected in Table 9.

	True Positive	True Negative
Question 1: Slots 1 – 20	100	100
Question 2: Slots 1 – 5	100	100
Question 3: Slot 1	100	100

Table 9 - Phase 1 Real World Data: Percentage of times patient (%) was found in the list by question

### 6.3 Phase 2: How much does the sensitivity of the algorithm fade as the data becomes more realistic?

In the second phase of this evaluation, I set out to determine the minimum data points necessary to identify a correct match. In real world situations, data are often incomplete, uncertain or missing. In this phase, the patient-matching algorithm was used to identify patients when the address is missing, age is incorrect, month of birth is estimated, day of birth is estimated, and month and day of birth are both estimated.

#### 6.3.1 Methods

To determine the degree to which the algorithm's accuracy fades as the data become more realistic, the number of data points (Section 6.2) the algorithm used to identify a match was systematically reduced.

My hypothesis is that if all available data points match, the algorithm's result will be a 100% match, and the probability of a correct match would decrease as the data points became more incomplete. For the purposes of this evaluation, incomplete can be either of the following:

- 1) Data fields in List A are complete, but the same data fields are missing in List B.
- 2) Data fields in List A are complete, but are transposed, or estimated (Age, Date Of Birth) in List B.

To test this hypothesis, all patients from the clinical visit and clinical research simulated dataset were used. This phase of the evaluation was conducted using the simulated dataset only. 549 patients from the clinical research dataset formed List A (Probe List). List B was comprised of patients from the clinical visit dataset, resulting in 1000 patients. List A is a clinical research dataset, having fewer personal identifying data points, only numbers 13 – 16 from Section 6.2. List B will mimic a clinical visit dataset, which contains more personal identifying data points, 1 – 16 from Section 6.2. The algorithm was run using both lists 5 times. List A stayed the same and List B changed for each Setting to mimic the types of data that would be found in a clinical research dataset. Table 10, below, describes the changes made to each list for each Setting and the reason for the change.

<b>Setting</b>	<b>List A (Clinical Research)</b>	<b>List B (Clinical Visit)</b>	<b>Occurrence</b>
1	No Changes	No Changes	Ideal Situation
2	No Changes	No Address	Some clinical research settings
3	No Changes	Setting 2 + Incorrect Age (+/- 1 year)	Low Resource Settings
4	No Changes	Setting 3 + DOB – Month changed to "01"	Unknown Month of Birth – Low resource settings
5	No Changes	Setting 4 + DOB – Day changes to "01"	Unknown Month and Day of Birth – Low resource settings

**Table 10 - Phase 2: Patient-Matching Algorithm Evaluation Settings and Occurrences**

The algorithm's results for each Setting were used to answer the three questions listed in Section 6.1 Evaluation Metrics. Additionally for Question 3, I recorded the probabilities and location for each probe patient to identify the point at which the patient's probability of having a 100% probability of a correct match changed.

### 6.3.2 Results

The results of this phase, Table 11 below, show that the probe patient was always first in the patient list for Settings 1 through 4. For Settings 5 only 99% of patients were first in the list.

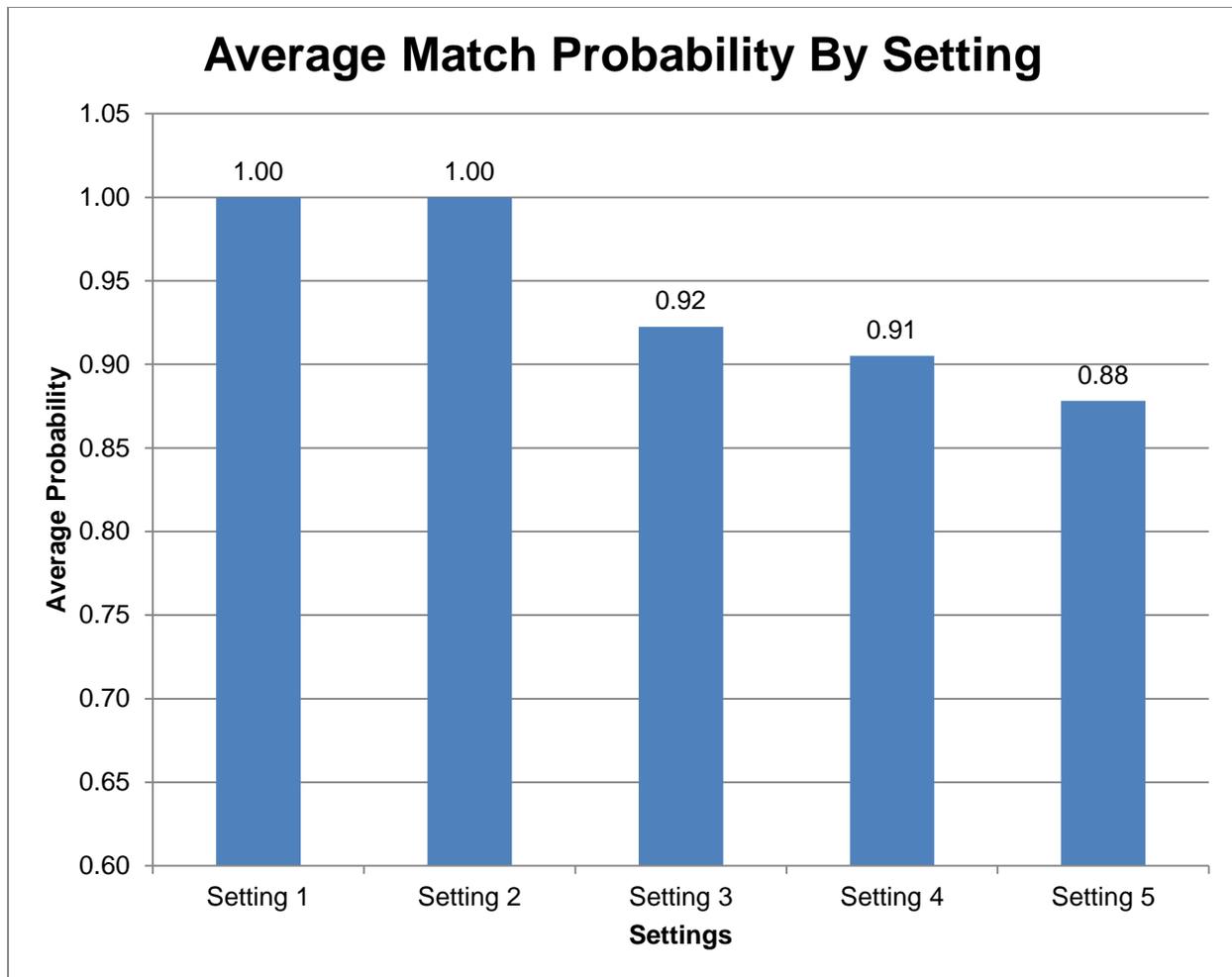
As seen in Table 12, the probability of a correct match decreased from 100% to 92% at Setting 3. The difference between Settings 3 and 4 was 1%. By Setting 5, the average probability of an correct match is 88%. This data is graphically represented Figure 29.

Setting Number	Question 1: Slots 1-20		Question 2: Slots 1 – 5		Question 3: Slot 1	
	True Positive	True Negative	True Positive	True Negative	True Positive	True Negative
1	100	100	100	100	100	100
2	100	100	100	100	100	100
3	100	100	100	100	100	100
4	100	100	100	100	100	100
5	100	100	100	100	99	100

Table 11 - Phase 2: Percentage of times patient (%) was found in the list by Setting number and question

Setting Number	Average Probability of Correct Match	Average Position of Correct Match
1	100	1
2	100	1
3	92	1
4	91	1
5	88	1

Table 12 - Phase 2: Average Probability and Position of Correct Match by Setting Number



**Figure 29 - Phase 2: Average Match Probability by Setting Number. Setting Number is shown on the X-axis, which the Y-axis shows the average probability.**

## 6.4 Conclusions

In Phase 1, given a threshold of the top 20 patients returned, using both simulated and real world data, the probe patient is always the first patient in the list, if a match existed. If no match existed, the patient did not appear in the list. Phase 1 was conducted using all 16 data points discussed in Section 6.2 of this Chapter. This indicates that the system is sufficient and the algorithm can correctly identify matching patients, given complete data.

Furthermore, in Phase 2 with given names, no address, and incomplete or uncertain age and date of birth, the matching patient was correctly identified by the algorithm 99% of the time. While the correct match was found, the probability of it being the correct match decreased. The difference between

Setting 2 (Address removed) and 3 (Address removed, and age altered) is significant, 8%; however, the difference between Setting 3 and 4 is 1%. This number decreased to 88% when both the month and date of birth are estimated for Setting 5.

The results of Phase 1 and Phase 2 show that the matching algorithm is sufficient to identify patients using both simulated and, and under some circumstances, real world data. Additionally, I have determined that complete data is not necessary to identify a match. Using this algorithm, the probe patient can be identified with no address, incorrect age, and unknown month and date of birth with an 88% average probability of a correct match.

While the results of Phase 1 are promising in that the probe patients is always first in the list, Phase 2 does raise questions of whether data quality needs to be improved or if the algorithm performs well enough to be used in a medical setting. While every effort should be made to provide high quality data, this is not always a practical scenario because of data transcription errors, or incorrect or incomplete information provided by the patient.

Only the Levenshtein algorithm with blocking based on gender was used during this evaluation. Algorithms such as longest substring and Jaro-Winkler and many others can be used instead of or in addition to those employed by this research to provide better results. Moreover, as this is medical data and any decision(s) made based on its use could potentially be life threatening, a human review of the algorithm's results should be conducted before actions are taken.

## Chapter 7: OBDIS Evaluation

This Chapter's focus is on the evaluation of the prototype system, developed in Chapter 4 (Table 6 and Figure 17). The prototype was evaluated using both types of data described in Chapter 5, and is based on the scenarios introduced in Chapter 4. Finally, a system adaptability and extensibility evaluation was conducted to determine the generality of my approach.

The purpose of this system is to provide the end user with a unified view of clinical research and clinical trials data from disparate sources. In order to be effective, the system should meet the requirements and scenarios of use described in Chapter 4. In addition, I set out to identify the answers to the following questions:

QN 1) Is the correct patient returned?

QN 2) Are the correct number of query results returned and are the results returned the same as those returned from the gold standard, the original systems OpenMRS and OpenClinica?

QN 3) Is the prototype system's response time less than that of the original systems combined?

QN 4) How general is the prototype system?

a) Adaptability: What changes to the prototype system are required if changes to the current ontology(ies) are made?

b) Adaptability: What changes to the prototype system are required if one or more of the current ontology(ies) are replaced with a new ontology(ies)?

c) Extensibility: How much work is required to add a new information system/data source?

Questions 1-3 are data-related and were answered using simulated and real world data. The remaining question (4) was answered during the informatics evaluation.

### 7.1 Methods

In an effort to evaluate the prototype system, two types of data were used -- simulated and real-world data, described in Sections 5.1 and 5.2 of Chapter 5. Ten percent of all patients from the clinical research and clinical visit datasets were randomly chosen as probe patients for Queries 1 – 3 (Table 13). This resulted in 140 patients from the simulated dataset and 472 patients from the real world dataset.

Queries 4 – 6 (Table 13) return cohorts and were evaluated using a scenario created by me. The results of these queries were totaled and compared to the gold standard.

The evaluation was automated using Java code. The gold standard was created by querying the original systems directly, using the randomly selected patient data query parameters (i.e. without the use of the ontology). Execution time was calculated based on the amount of time required by the respective databases to execute the query. Code from the prototype system was used to determine how the system works compared to the gold standard. Based on the results from the patient-matching algorithm evaluation from Chapter 6, I assumed the correct patient was always number one in the list of patients returned.

### **7.1.1 Scenarios Of Use**

For each data type, simulated and real world, three categories of questions were asked: Patient Identifier Information, Clinical Trial Cohorts, and Patient Follow-Up. Each of these categories are represented in the scenarios of use introduced in Chapter 4, Table 6. Five scenarios resulted from the work in Chapter 4. For evaluation purposes, the Scenarios of Use were further divided into the seven queries listed in Table 13 below. I believe these queries are representative of the types of queries required of a data integration system in this setting. Queries 1-6 were evaluated using the original systems as the gold standard and the experimental system as the variable. The remaining query (7) is associated with Scenario 4, which was not implemented in the prototype system and therefore not involved in the evaluation.

Scenario	Query Number	Query
S1	Q1	Find all data for Patient X.
S1	Q2	Find all future visit dates for Patient X.
S1	Q3	Identify recent laboratory data for Patient X.
S2	Q4	Determine the inclusion criteria for a clinical trial, and query systems to determine patients who will be included or excluded from the trial.
S3	Q5	Find all patients of Physician Y who are on ART and have not had a clinical encounter in the past X months.
S3	Q6	Find all patients of Clinic Z on drug regimen ABC and DEF during time period G.
S4	Q7*	Transfer patient data from data source A to data source B.

**Table 13 - Scenarios of Use by Query Number (\* Query 7 was not implemented in the prototype system.)**

### 7.1.2 Evaluation Metrics

Six metrics, listed in Table 14 below, were used to evaluate the prototype system and were calculated by hand. These metrics correspond to Questions 1-3, presented at the beginning of this chapter, and the Queries from Table 13.

Metric Number	Metric
M1	Response time
M2	# Patient Encounters
M3	# Scheduled visits
M4	# Laboratory data points
M5	# Patients Returned
M6	Compare Patient Name, DOB, Gender

**Table 14 - Metrics Used to Evaluate Prototype System**

To determine whether the correct patient was returned, Question 1, Metric M6, the probe patient's name, date of birth and gender were used. To determine whether the correct number of query results was

returned, Question 2, the total number of: patient encounters, scheduled visits, laboratory data points and patients returned were used, Metrics 2 – 5 respectively. Timeliness, Question 3, was determined by calculating the time from query submission to results returned, Metric M1. Table 15 provides a listing of all Questions, Queries and the Metrics used to evaluate them. Crossing columns, Questions, and rows, Queries, will provide the associated Metric.

Query	Question		
	QN1	QN 2	QN 3
Q1	M6	M2	M1
Q2	M6	M3	M1
Q3	M6	M4	M1
Q4	M6	M5	M1
Q5	M6	M5	M1
Q6	M6	M5	M1

**Table 15 - List of Queries, Questions and Associated Metrics**

These results were used to calculate precision (Equation 1) and recall (Equation 2) for each question. These estimates are often used in the domain of information retrieval. Precision is defined as “an estimate of the conditional probability that an item will be relevant given that it is retrieved” (114). Alternatively, recall is “an estimate of the conditional probability that an item will be retrieved given that it is relevant” (114).

$$Precision = \frac{\# \text{ correct results retrieved}}{\# \text{ of all results retrieved}} \quad \text{Equation 1}$$

$$Recall = \frac{\# \text{ correct results retrieved}}{\# \text{ results that should have been retrieved}} \quad \text{Equation 2}$$

## 7.2 Simulated Data Evaluation Results

Tables 16 – 28, below, provide a view of the accuracy: total number of results returned per query for both the gold standard and the prototype system; and average response time for each query. The prototype system returned the same number of results as the gold standard for queries 2, 4, 5 and 6. For queries 1 and 8, the gold standard returned fewer queries than the prototype system. When compared, patient name, date of birth and gender were found to be the same for both the original and prototype systems when the same number of results were returned.

Response time was measured in seconds, and across all queries, the prototype system took longer to return results than the original systems combined. On average, the prototype system took 63.1 seconds and the original systems combined 4.98 seconds, a difference of 58.12 seconds per query.

Scenario 1: Querying other provider's systems to identify prior medical records for a given patient.

Scenario 1, Query 1: Find all data for Patient X.

Metrics used: #1 (Response time), #2 (# Patient Encounters), #6 (Compare Patient Name, DOB, Gender)

	Total
Original System	1.52
Prototype	41.36

Table 16 - Query 1: Average time required to return results (in seconds)

	OpenClinica	OpenMRS	Total
Original System	276	380	656
Prototype	292	380	672

Table 17 - Query 1: Total number of patient encounters

Query 2: Find all future visit dates for Patient X.

Metrics used: #1 (Response time), #3 (# Scheduled visits after given date), #6 (Compare Patient Name, DOB, Gender)

Visit Date after January 1, 2011

	Total
Original System	1.13
Prototype	40.5

Table 18 - Query 2: Average time required to return results (in seconds)

	Total
Original System	22
Prototype	22

Table 19 - Query 2: Total number of scheduled visits after January 1, 2011

Query 3: Identify recent laboratory data for Patient X.

Metrics used: #1 (Response time), #2 (Laboratory data points), #6 (Compare Patient Name, DOB, Gender)

Query Parameters: Observation date: January 1, 1990 – January 1, 2013; Observations: CD4;

Total Lymphocyte Count

	Total
Original System	1.23
Prototype	38.72

Table 20 - Query 3: Average time required to return results (in seconds)

	Total
Original System	184
Prototype System	276

Table 21 - Query 3: Total number of laboratory data points

Scenario 2: Determining the inclusion criteria for clinical trials, and querying systems to determine patients who will be included or excluded from the trial.

Query 4: Identify patients who will be included in a clinical trial

Metrics used: #1 (Response time), #5 (Patients Returned), #6 (Compare Patient Name, DOB, Gender)

Gender: Male & Female

Age: 20 – 30

Patient Status: Alive

WHO Stage: Adult 2

Date: January 1, 1996 – December 31, 2012

	OpenClinica	OpenMRS	Total
Original System	1.0	5.0	6.0
Prototype			83.0

Table 22 – Query 4: Average time required to return results (in seconds)

	OpenClinica	OpenMRS	Total
Original System	62	8	70
Prototype	62	8	70

Table 23 - Query 4: Total number of patients returned

Scenario 3: Identifying adverse events, routine monitoring, side effects or contraindications caused by or required by routine clinical care or clinical trial study protocols.

Query 5: Find all patients of Physician Y who are on ART and have not had a clinical encounter in the past X months.

Metrics used: #1 (Response time), #5 (# Patients Returned), #6 (Compare Patient Name, DOB, Gender)

Physician Y: Brian Wilson

Patient Status: Alive

Drug: 1a(30) Stavudine (30) Lamivudine/Nevirapine

Visit Date: December 31, 2001 - July 1, 2011

	OpenClinica	OpenMRS	Total
Original System	5.0	9.0	14.0
Prototype			65.0

Table 24 - Query 5: Average time required to return results (in seconds)

	OpenClinica	OpenMRS	Total
Original System	549	484	1033
Prototype	549	484	1033

Table 25 - Query 5: Total number of patients returned

Query 6: Find all patients of Clinic Z, on drug regimen ABC and DEF during time period G.

Metrics used: #1 (Response time), #5 (# Patients Returned), #6 (Compare Patient Name, DOB, Gender)

Clinic: AAR Nakuru Clinic

Patient Status: Alive

Drug: Zidovudine/Lamivudine/Efavirenz; D4T(30)/3TC/NVP; D4T(40)/3TC/EFV

Visit Date: January 1, 2000 – December 1, 2012

	OpenClinica	OpenMRS	Total
Original System	1.0	5.0	6.0
Prototype			110.0

Table 26 - Query 6: Average time required to return results (in seconds)

	OpenClinica	OpenMRS	Total
Original System	0	4	4
Prototype	0	4	4

Table 27 - Query 6: Total number of patients returned

Table 28 below provides the precision and recall for each query. This table is consistent with the results from Tables 17, 19, 21, 23, 25 and 27 in that precision and recall equal 1, for all queries.

Query	Precision	Recall
Q1	0.98	1
Q2	1	1
Q3	1	0.67
Q4	1	1
Q5	1	1
Q6	1	1

**Table 28 - Simulated Data Precision, Recall and f-Measure by Query**

### 7.3 Real World Data Evaluation Results

Tables 29 - 41 below provide a view of the accuracy: number of results returned per query for both the gold standard and the prototype system; and average response time for each query. The results for queries 4, 5 and 6 were the same for both the prototype and the gold standard. When patient name, date of birth and gender were compared, the results were found to be the same for both the original and prototype system. The manual system returned fewer results than the prototype system for queries 1 – 3.

Response time was measured in seconds and across all queries, the prototype system took longer to return results than the original systems combined. On average, the prototype system took 46.86 seconds and the original systems took a combined 5.7 seconds, a difference of 41.12 seconds per query.

Scenario 1: Querying other provider's systems to identify prior medical records for a given patient.

*Query 1: Find all data for Patient X.*

*Metrics used: #1 (Response time), #2 (Number of Patient Encounters), #6 (Compare Patient Name, DOB, Gender)]*

	Total
Original System	0.03
Prototype	31.75

Table 29 – Query 1: Average time required to return results (in seconds)

	OpenClinica	OpenMRS	Total
Original System	1231	5300	6531
Prototype	4056	5939	9995

Table 30 - Query 1: Total number of patient encounters

Query 2: Find all future visit dates for Patient X.

Metrics Used: #1 (Response time), #3 (#Scheduled visits after given date), #6 (Compare Name, DOB, Gender)

Query Parameter: Visit Date after January 11, 2011

	Total
Original System	0.005
Prototype	32.33

Table 31 - Query 2: Average time required to return results (in seconds)

	Total
Original System	1596
Prototype	2499

Table 32 - Query 2: Total number of scheduled visits after January 11, 2010

Query 3: Identify recent laboratory data for Patient X.

Metrics used: #1 (Response time), #4 (#Laboratory data points), #6 (Compare Patient Name, DOB, Gender)

Query Parameters: Observation date: January 1, 1990 – January 1, 2013; Observations: Height, Weight, Mid-Upper Arm Circumference

	Total
Original System	0.15
Prototype	31.1

Table 33 - Query 3: Average time required to return results (in seconds)

	Total
Original System	10265
Prototype	14812

Table 34 - Query 3: Total number of laboratory data points

Scenario 2: Determining the inclusion criteria for clinical trials, and querying systems to determine patients who will be included or excluded from the trial.

Query 4: Identify patients who will be included in a clinical trial

Metrics used: #1 (Response time), #5 (#Patients Returned), #6 (Compare Patient Name, DOB, Gender)

Age: 20 – 30

Patient Status: Alive

Diagnosis: Herpes Zoster

Date: January 1, 1990 – December 31, 2011

	OpenClinica	OpenMRS	Total
Original System	10.0	9.0	19.0
Prototype			87.0

Table 35 – Query 4: Average time required to return results (in seconds)

	OpenClinica	OpenMRS	Total
Original System	227	6	233
Prototype	227	6	233

Table 36 - Query 4: Total number of patients returned

Scenario 3: Identifying adverse events, routine monitoring, side effects or contraindications caused by or required by routine clinical care or clinical trial study protocols.

Query 5: Find all patients of Physician Y who are on ART and have not had a clinical encounter in the past X months.

Metrics used: #1 (Response time), #5 (# Patients Returned), #6 (Compare Patient Name, DOB, Gender)

Physician Name: tchonga

Drug: Zidovudine

Visit Date: May 5, 2005 – May 5, 2009

	OpenClinica	OpenMRS	Total
Original System	12.0	1.0	13.0
Prototype			56.0

Table 37 - Query 5: Average time required to return results (in seconds)

	OpenClinica	OpenMRS	Total
Original System	971	0	971
Prototype	971	0	971

Table 38 - Query 5: Total number of patients returned

Query 6: Find all patients of Clinic Z, on drug regimen ABC and DEF during time period G.

Metrics used: #1 (Response time), #5 (# Patients Returned), #6 (Compare Patient Name, DOB, Gender)

Clinic: KEMRI Wellcome Trust Kilifi

Drug: Amoxicillin

Visit Date: July 1, 1983 – December 1, 2001

	OpenClinica	OpenMRS	Total

<b>Original System</b>	1.0	1.0	2.0
<b>Prototype</b>			43.0

**Table 39 - Query 6: Average time required to return results (in seconds)**

	<b>OpenClinica</b>	<b>OpenMRS</b>	<b>Total</b>
<b>Original System</b>	0	0	0
<b>Prototype</b>	0	0	0

**Table 40 - Query 6: Total number of patients returned**

Table 41 below provides the precision and recall each query. This table is consistent with the results from Tables 32, 34, 36, 38 and 40 in that precision and recall equal 1.

<b>Query</b>	<b>Precision</b>	<b>Recall</b>
<b>Q1</b>	1	0.65
<b>Q2</b>	1	0.64
<b>Q3</b>	1	0.69
<b>Q4</b>	1	1
<b>Q5</b>	1	1
<b>Q6</b>	1	1

**Table 41 - Real World Data Precision, Recall and *f*-Measure by Scenario and Query**

## **7.4 Adaptability and Extensibility Evaluation**

The author developed the prototype system, with input from Drs. Abernethy, Brinkley and Walson. However, clinicians and researchers in low-income settings were not used to provide insight into pertinent design decisions. Therefore, the results of this evaluation are based upon the author's assessment of the system and its functionality.

The final question, "How general is the system?" was answered during this evaluation.

*QN 4: How general is the system?*

*QN 4 a: Adaptability: What changes to the system are required if changes to the current ontology(ies) are made?*

The system was developed to allow the user to interchange systems and ontologies with minimal programming knowledge and effort. Therefore, making changes to the current ontologies only requires the user make changes to the system configuration: ontology to DB, and ontology to UI mapping documents. Based on the extent of the ontology changes, updating the required documents could take as little as an hour.

The ontologies used in this work were created using an iterative approach, which happened while the prototype system was being developed. During the second and third settings of the PtID Ontology, extensive changes to the ontology were made and it took approximately two hours to update the appropriate documents. The HIV Ontology is larger than the PtID Ontology and updating documents based on iterative changes required four to six hours. Updating these documents requires basic knowledge of XML, and a working knowledge of ontologies. A deficit in either of these areas could increase this time.

The amount of time required to make changes to these documents can be reduced if the databases to be integrated do not use the object relational model employed by both OpenMRS and OpenClinica. Flat database models are easier to map because concept IDs, like those in OpenMRS, are not required. In addition to mapping documents, the current SPARQL queries should be reviewed and tested for accuracy. The current SPARQL queries are based on the structure of the current ontologies and could return incorrect information should the ontologies change.

*QN 4 b: Adaptability: What changes to the system are required if one or more of the current ontology(ies) are replaced with a new ontology(ies)?*

In addition to the mapping document and SPARQL query changes, suggested in Q4 a, U.I.s need to be changed as well. If the HIV/AIDS ontology is replaced with a different ontology, the U.I. for the Cohort and Scenario-Based Query screens should be changed to reflect the new subject matter. These interfaces incorporate hard-coded drop down and auto complete lists, which are specific to HIV/AIDS, its

associated opportunistic infections, disease states and medications, and should be altered to reflect the new ontology(ies) subject matter.

*QN 4 c: Extensibility: How much work is required to add a new system/data source?*

The prototype system was built using the simulated datasets. To transition to the real world datasets, new mapping documents were created. The SPARQL and XQuery queries did not change. Additionally, because of the system's design, the SQL queries are automatically generated based on the user's input and the results of the SPARQL queries and therefore, no work on the author's part was necessary. However, the user should verify that the correct results are being returned.

## 7.5 Discussion

In this section the results above are reviewed, organized by the research questions each study addressed.

*QN 1: Is the correct patient returned?*

*QN 2: Are the correct number of query results returned and are the returned results the same as those returned from the gold standard?*

As the results from Section 7.2 and 7.3 show, the Prototype system did not always return the correct number of patients when specific patient information was entered as query parameters. This can be attributed to the assumption that the first patient returned was the query patient. Alternatively, the correct number of patients were returned when cohort queries were submitted, as compared to the gold standard. Manual comparison of the patient identifying information for each patient returned confirmed that the same patients were returned from the prototype system as compared to the gold standard.

To further confirm that the results returned were comparable, the prototype system-generated queries were compared to the author's manually created queries for each query. While the queries are structured differently, semantically the queries are the same. A listing of the generated and manually created SQL queries by Scenario and Query can be found in Appendices J and K, Simulated and Real World, respectively.

*QN 3: Is the system's response time less than that of the original systems combined?*

The amount of time required to process a query in the prototype system is significantly higher. During query processing, a progress meter is shown to inform the user of the time left in completion of their query. While research has shown that feedback such as this does alleviate the effects of user satisfaction drop, the wait time is still significantly higher than the 3-12 seconds often used in web site delay research (115,116).

The time difference can be attributed to the difference in the number of queries required to provide results. Using the original systems, OpenMRS or OpenClinica, only one database query is executed. Alternatively, the prototype system requires four queries: two XQueries; one SPARQL; and one SQL DB query, depending on query type. Clicking on a Patient's Clinic ID to view Patient Encounter History would not require the use of Patient ID associated ontology queries and would therefore result in one less XQuery and SPARQL query. More efficient queries can be developed, which could reduce the amount of time required to process query results; however, the number of queries would not decrease. Finally, the processing time required of the patient-matching algorithm also adds to the prototype system's response time. This algorithm is not run in the individual systems. However, it should be noted that the prototype's queries have not been optimized, which could decrease the time required to process queries. Furthermore, the patient populations used to evaluate the system are considered small by medical standards and larger populations would further increase query processing time.

*QN 4: How general is the prototype system?*

The prototype system can be adapted, changes to the current ontologies or new ontologies with minimal developer knowledge and effort. This is especially important as ontologies are considered "living documents" in that they are subject to changes based on the needs of the user. While extending the system to incorporate new data sources does require developer knowledge to update the user interface, as needed, much of the backend, query development, is automated and does not need changing.

## **7.6 Conclusions**

In this chapter, I set out to answer four questions related to the functionality provided by the prototype system. Of the data returned for both simulated and real world data, the correct information

was returned although not all expected data was returned. This can be attributed to the assumption that the first patient returned matched the probe patient.

Additionally, the system did not perform as expected as it relates to the amount of time required to execute queries. While this could deter usage, I believe that the ability to view data from multiple sources in one aggregate view will outweigh the need to wait longer for query results.

In addition to the functionality provided by the system, I believe that the ability to make changes to the ontologies and data sources used by the system with minimal effort and programming knowledge will further encourage adoption of this system to integrate disparate data sources.

## Chapter 8: Conclusions

### 8.1. Limitations

#### 8.1.1 Limitations in Data Collection and Ontology Development

The data collected in Chapter 3 was a convenience sample and, therefore, not as comprehensive as I would have liked. Access to a more robust sample, e.g. all data collection forms used in Kenyan HIV clinics would have provided a better understanding of the types of data collected throughout the country. Moreover, this data would have also provided better insight to the types of diseases and treatments that are used throughout the country. This is especially important because of diseases such as Malaria, which are endemic to certain parts of the country and not others.

Another limitation of this study is the exclusion of clinical vocabularies or terminologies. Resources such as the Unified Medical Language System (UMLS) metathesaurus [15] and the National Cancer Institute (NCI) metathesaurus link several existing vocabularies that may contain several concepts relevant to this domain. Clinical vocabularies, while not initially created to serve the specialized needs of developing countries, are increasingly being used in this setting; hence, their coverage of patient identifiers may become increasingly relevant. Similarly, our study did not include data exchange standards such as Health Level Seven (HL7) [16] or semantic web data models, both of which will conceivably be used with a higher frequency as collaborative sites begin to share more data. More generally, a broader search for relevant data models and ontologies would bolster our results. A more comprehensive analysis of systems and standards in use would improve our chances of realizing a standardized model.

Only Drs. Abernethy, Brinkley and Walson reviewed the ontologies created with this work. Because the ontology was not reviewed by external domain experts or ontology experts, information important to the validity and soundness of the ontology as well as the workflow of the intended user base, could have been excluded.

#### 8.1.2 Limitations in Integration Methods

The prototype system uses OpenClinica and OpenMRS. While these systems are used

throughout the world, they are not the only open source systems of their kind available. More importantly, many homegrown systems are in use and have not been included in this study. Therefore, this work is limited in that the systems used for integration may not be generalizable to all systems that are in use.

This work implemented a database federation with a mediated schema data integration method. This method allows for data to be stored locally, and therefore access can be controlled by its owner. More importantly, all data sources are mapped to a single ontology. This reduces the need for end users to learn multiple schema before querying a source. While this approach does meet the needs of our proposed user base, a hybrid approach with a data warehouse model might allow the user more flexibility in the types of data they are able to access without a synchronous data connection.

### **8.1.3 Limitations in Evaluation**

The evaluation of the system was carried out by the author and did not include a usability study. This study could have provided insight into the workflow of the proposed user base, aid in making the user interface and system configuration more user friendly, and encourage adoption.

## **8.2. Future Work**

### **8.2.1 Save Functionality**

The current version of the cohort query provides a list of patients that meet the search criteria. However, the ability to export or save the results or to save user-defined queries to a personal computer or in the system would allow the researcher to reuse queries and results. This functionality might be important for repeatability of clinical trials, participant recruitment, or other use cases.

### **8.2.2 User Interface**

#### **8.2.2.1 Cohort Query Flowchart Interface**

The cohort query screen would also benefit from a flowchart interface. This interface would allow the user to visually see how their criterion changes their results. *Ex. Find all patients with a WHO stage of 1 or 2, age > 25, on 3TZ.* The system, in its current form, would provide you with the result in the format shown in Figure 30 below. Making changes (age > 35) to the query would require the user to return to the query form, edit the query, and submit the updated query. The user is not privy to how

requiring the age requirement from age > 25 to age <35, or moving age higher in the list of criterion changes their results other than reviewing the number of patients returned.

[Pt ID Query Cohort Query Scenario-Based Queries About Logout](#)

### Cohort Query Results

Query Parameters: Gender: Male & Female; [Edit Query](#)

ClinicID	Name Prefix	First Name	Middle Name	Last Name Prefix	Last Name	Last Name 2	Last Name Suffix	Gender	Birthdate	Age*	Status	Clinic ID Matches	Probability	Show Comparison
<a href="#">57567</a>		Edward			Kipchirchir			M	21-11-1988	24				
<a href="#">41513</a>		Maria			Moseti			F	26-04-1991	21				
<a href="#">69473</a>		Maria			Yusuf			F	12-08-1982	30				
<a href="#">53560</a>		Isla			Morong			F	26-04-1986	26				
<a href="#">61026</a>		Gabriel			Libwab			M	21-06-1991	21				
<a href="#">55345</a>		Arthur			Sambai			M	14-09-1985	27				
<a href="#">52488</a>		Lucy			Andala			F	17-08-1989	23				
<a href="#">43124</a>		Scarlett			Kabiroi			F	15-06-1991	21				
<a href="#">10</a>								m	30-08-1989	23				
<a href="#">30</a>								m	21-11-1988	24				
<a href="#">38</a>								m	25-08-1987	25				
<a href="#">39</a>								m	10-08-1983	29	2392	0.5	<a href="#">2392</a>	
<a href="#">68</a>								m	26-04-1991	21				
<a href="#">81</a>								m	12-08-1982	30				
<a href="#">84</a>								m	20-07-1990	22				
<a href="#">90</a>								m	07-08-1989	23				
<a href="#">97</a>								m	23-07-1984	28				
<a href="#">101</a>								m	08-02-1990	22				

**Figure 30 - Cohort Query Results Screenshot**

A flowchart interface would provide the criteria results breakdown (WHO Stage 1 or 2 c 5000 of 6000, age 3,000 of 5000 and 3tz 1500 of 3000), but also allows the user to change the search criteria by dragging and dropping criteria based on priority.

### 8.2.2.2 Patient Matching Results

The current query results screen shows the user the clinic ID and the probability returned by the matching algorithm with probabilities greater than 50%. In an effort to reduce information overload, for probabilities less than 50%, no information is shown. While I believe this to be enough information to allow the user to make choices as to whether these patients could in fact be the same person, the ability

to see all matches, potential matches and non-matches or any combination of the three would give the user more or less information should they so choose.

### 8.2.3 Configuration

While the prototype system is fully functional, more work should be done to make configuring the system more user friendly. In chapter 4, I discussed the components of the system, which implementers would need to configure, in order to integrate their data sources and ontologies. While most of this work cannot be avoided no matter how much automation is incorporated, automation can make the task less daunting and time consuming.

Currently, there is no user interface for configuration procedures, specifically, developing mapping documents and configuring the matching algorithm. Allowing the user to visually see database tables and columns, opposite the ontology classes and properties, could help to reduce confusion and information overload. Automatically generating the appropriate XML configuration files based on user input would also eliminate the implementers XML knowledge requirement.

The DB to Ontology mapping document includes the option to use API calls to query data sources; however, this functionality is not fully implemented. A complete implementation of the functionality would make the system more robust and could potentially reduce the workload of the system, making it more efficient and thereby reducing the amount of time required to process queries and return results.

The current matching algorithm only uses the Levenshtein distance metric to calculate the probability of two patients being the same person. The Java libraries (simmetrics and duke) that are used by the system provide other metrics, which can be used in addition to or instead of the Levenshtein formula. Moreover, because of the knowledge required to choose and configure the algorithms to return accurate data, the configuration process was hard coded. Allowing implementers to configure the algorithm for themselves through a user interface would afford the implementer the ability to fine tune results to their dataset. This will become more important when algorithms that provide name matching such as Soundex are created or expanded to include African dialects.

#### **8.2.4 Ontology Refinement**

Working with domain and ontology experts would help to better refine the ontology and provide insight into more efficient ways to represent and query the data. These changes could also help to reduce the amount of time required to return query results.

#### **8.2.5 Data Transfer between systems**

Currently, the system is view only; users can see data, but cannot update, add or transfer data between systems. Allowing the user to transfer data between systems would reduce the need to erroneously enter data and help to reduce redundancy in workflows and clinical procedures.

#### **8.2.6 Non-database data sources**

Only database data sources are allowed in the current prototype implementation. However, functionality which allows for CSV or XML based data would not only reduce the need for gaining access to an institution's database, but also allow the user to have access to the data when always on connections are unavailable, which is often the case in low-resource settings.

### **8.3. Deployment Recommendations**

#### **8.3.1 Deployment using a single server with network access to multiple remote sites**

The implementation described in this work can be deployed as long as access to the databases in question is available.

#### **8.3.2 Deployment using no server and multiple remote sites**

If no server is available and databases are maintained on personal computers, it is my recommendation that data be exported from the databases and imported into separate databases on a single computer and deployed as is. This implementation will reduce the amount of time required to communicate over a data network.

However, if a data network is preferred, access to the IP addresses of the databases should be included in the system configuration file and the system deployed as is. This approach will increase the amount of time required to return query results to the user because the system will have to communicate and transfer data over the data network.

## 8.4. Contributions

This research provides two ontologies, HIV and Patient Identifier. These ontologies are useful for the fields of bioinformatics and medicine as they provide standardization of concepts that are often used in these fields to describe data, protocols and actions performed by both humans and information systems. Additionally, the Patient Identifier Ontology is useful to the field of data linkage, which is often associated with statistics and database systems. While parts of these ontologies do exist from disparate sources, it is our belief that comprehensive versions, such as those presented here, are not available. Additionally, ontologies developed using the methods from Chapter 3 do not exist.

Moreover, I have presented an open source solution, which can be helpful in Kenya during and after their transition to a country-wide electronic medical record system. Specifically, the OBDIS can act as a bridge between the new and old systems until time permits or instead of performing extract, transform and load procedures, which can be expensive and time consuming.

The OBDIS is unique in that it automatically generates SQL queries that are run over the respective databases. This functionality reduces the need for custom programming, a practice, which can be time consuming and results in legacy systems that are difficult to maintain.

Additionally, I offer a lightweight solution that does not require network access and is functional with both MySQL and Postgres database systems and two open source systems OpenMRS and OpenClinica, popular throughout the world.

I believe this system can be useful in both the current pre-standard unified EMR setting in Kenya and the post-standard unified EMR setting. The current health information system climate with its disparate homegrown and open source systems would benefit from the standards set forth by the ontology as well as the system's ability to provide access to data that is currently available. After the adoption of the new EMR system, Kenya could utilize this system to provide access to the data in legacy systems during and well after the transition.

While the unified EMR's purpose to provide a standard for collecting medical data in Kenya's MOH run facilities, non-MOH run facilities may continue to silo their data by choice or based on regulations set forth by their funding agency. This research could provide an avenue by which they can

share data with the MOH and not implement their system in order to provide better care for all patients involved.

More importantly, as discussed in Chapter 6, this research can provide a means for furthering the dialogue around data quality and record linkage. While both topics have been discussed and are the subject of research studies throughout the world, providing access to data such as this makes that work more important. Specifically, when sharing data emphasis lies on whether the data can be trusted, are the data fields accurate and up to date, how often are laboratory machines calibrated and serviced. Inaccurate data or uncalibrated machines do not provide useful data; instead, they provide data that can result in harmful patient outcomes.

Most record linkage algorithms were developed for English dialects and have not been adapted for African dialects. The need to identify patients across multiple systems and large patient populations will increase and require algorithms that more accurately identify matching patients.

While the original intended use of the prototype system was for use in low resource systems, this work is useful for and can be employed in resource-rich environments as well. This is especially relevant in the U.S. with the increased funding for meaningful use compliance and Health Information Exchanges (117). Both initiatives push the healthcare agenda towards an increase in data sharing, which like low-income countries is deterred by siloed systems.

## 8.5 Summary

In completing this work, I set out to provide an open source data integration solution that met the seven requirements discussed in Chapter 4:

- R1. The solution must have little or no implementation cost and be maintainable by the local population.
- R2. The solution must be compatible with currently deployed software.
- R3. The solution must be functional with asynchronous data connectivity.
- R4. The solution must be able to answer questions posed by clinicians and researchers.
- R5. The solution must be flexible enough to incorporate multiple diseases and medical conditions.

R6. The solution must be usable by non-informaticists or IT personnel.

R7. The solution must provide information in a timely manner.

Based on the results provided in Chapters 6 and 7, I have accomplished this task and provided a prototype system that meets these requirements.

I have provided an overview of the types of data and information that are created and used related to the treatment and care of HIV/AIDS patients in low-resource settings. I have also provided a solution, which can help to bridge the gap between siloed systems and allow for secondary use of clinical research data, an effort that has been championed by the meaningful use standards being employed in the U.S. More importantly, this system provides access to data, which could help to inform medical decision-making and reduce the need for duplicate and unnecessary tests.

## References

1. Ministry of Medical Services, Ministry of Public Health and Sanitation. Kenya National Health Accounts 2009/10 [Internet]. Available from: [http://www.who.int/nha/country/ken/kenya\\_nha\\_2009-2010.pdf](http://www.who.int/nha/country/ken/kenya_nha_2009-2010.pdf)
2. National AIDS and STI Control Programme. The Kenya AIDS Epidemic Update 2011 [Internet]. Available from: [http://www.unaids.org/en/dataanalysis/knowyourresponse/countryprogressreports/2012countries/ce\\_KE\\_Narrative\\_Report.pdf](http://www.unaids.org/en/dataanalysis/knowyourresponse/countryprogressreports/2012countries/ce_KE_Narrative_Report.pdf)
3. Kenya: Dr Ndemo, Dr Getao to push adoption of effective Tele-medicine at AFRIHEALTH Conference [Internet]. NGO News Africa. [cited 2013 Mar 22]. Available from: <http://ngonewsafrika.org/archives/10775>
4. Standards and Guidelines for Electronic Medical Record Systems in Kenya.
5. WHO | HIV/AIDS [Internet]. [cited 2011 Oct 28]. Available from: [http://www.who.int/topics/hiv\\_aids/en/](http://www.who.int/topics/hiv_aids/en/)
6. WHO | HIV/AIDS [Internet]. 2011 [cited 2011 Oct 28]. Available from: [http://www.who.int/topics/hiv\\_aids/en/](http://www.who.int/topics/hiv_aids/en/)
7. Guidelines for Antiretroviral Therapy in Kenya. Nairobi, Kenya: National AIDS/STI Control Program (NAS COP); 2011. Report No.: 4th Edition.
8. (UNAIDS) JUNP on H. Global Report: UNAIDS Report on the Global AIDS Epidemic. 2010.
9. Guidelines for Prevention and Treatment of Opportunistic Infections in HIV-Infected Adults and Adolescents [Internet]. Centers for Disease Control and Prevention; Available from: [http://aidsinfo.nih.gov/contentfiles/Adult\\_OI.pdf](http://aidsinfo.nih.gov/contentfiles/Adult_OI.pdf)
10. WHO | Kenya [Internet]. 2010 [cited 2011 Oct 28]. Available from: <http://www.who.int/countries/ken/en/>
11. Nations U, Assembly G. NATIONAL AIDS CONTROL COUNCIL UNGASS 2010. 2010.
12. The PIH Guide to the Community-Based Treatment of HIV in Resource-Poor Settings [Internet]. Partners In Health; 2008. Available from: [http://parthealth.3cdn.net/e0d6bb33c14b95300c\\_82m6b4ooc.pdf](http://parthealth.3cdn.net/e0d6bb33c14b95300c_82m6b4ooc.pdf)
13. HIV/AIDS: More proof that PrEP works [Internet]. IRINnews. [cited 2013 Mar 23]. Available from: <http://www.irinnews.org/report.aspx?reportid=93226>

14. Anglemeyer A, Rutherford GW, Horvath T, Baggaley RC, Egger M, Siegfried N. Antiretroviral therapy for prevention of HIV transmission in HIV-discordant couples. status and date: New search for studies and content updated (no change to conclusions), published in. 2013;(4).
15. Noor AM, Alegana VA, Gething PW, Snow RW. A spatial national health facility database for public health sector planning in Kenya in 2008. *International Journal of Health Geographics* [Internet]. 2009 Mar 6 [cited 2013 Jan 7];8(1):13. Available from: <http://www.ij-healthgeographics.com/content/8/1/13>
16. Ministry of Medical Services, Ministry of Public Health and Sanitation. Kenya Service Provision Assessment Survey 2010 [Internet]. Available from: <http://www.measuredhs.com/pubs/pdf/SPA17/SPA17.pdf>
17. Devolution of Healthcare Services in Kenya: Lessons learnt from other countries [Internet]. KPMG Services Limited; 2013. Available from: <http://www.kpmg.com/Africa/en/IssuesAndInsights/Articles-Publications/Documents/Devolution%20of%20HC%20Services%20in%20Kenya.pdf>
18. Organization WH. WHO International Clinical Trials Registry Platform. 2010.
19. Kenya Facts and Figures 2009. Kenya National Bureau of Statistics; 2009.
20. AMPATH-Kenya Home | AMPATH-Kenya [Internet]. [cited 2013 Jan 23]. Available from: <http://www.ampathkenya.org/>
21. Tierney WM, Achieng M, Baker E, Bell A, Biondich P, Braitstein P, et al. Experience implementing electronic health records in three East African countries. *Stud Health Technol Inform*. 2010;160(Pt 1):371–5.
22. Faces Kenya | Family AIDS Care and Education Services [Internet]. [cited 2013 Jan 23]. Available from: <http://www.faces-kenya.org/>
23. Millennium Villages [Internet]. Millennium Villages. [cited 2013 Jan 23]. Available from: <http://www.millenniumvillages.org/>
24. Kanter AS, Negin J, Olayo B, Bukachi F, Johnson E, Sachs SE. Millennium Global Village-Net: bringing together Millennium Villages throughout sub-Saharan Africa. *Int J Med Inform*. 2009 Dec;78(12):802–7.
25. OpenMRS » Open source health IT for the planet [Internet]. [cited 2012 May 1]. Available from: <http://openmrs.org/>
26. Were MC, Emenyonu N, Achieng M, Shen C, Ssali J, Masaba JPM, et al. Evaluating a scalable model for implementing electronic health records in resource-limited settings. *J Am Med Inform Assoc* [Internet]. 2010 May [cited 2013 Jan

- 23];17(3):237–44. Available from:  
<http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2995711/>
27. Mohammed-Rajput NA, Smith DC, Mamlin B, Biondich P, Doebbeling BN. OpenMRS, A Global Medical Records System Collaborative: Factors Influencing Successful Implementation. *AMIA Annu Symp Proc* [Internet]. 2011 [cited 2013 Jan 23];2011:960–8. Available from:  
<http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3243141/>
  28. Fraser HS, Thomas D, Tomaylla J, Garcia N, Lecca L, Murray M, et al. Adaptation of a web-based, open source electronic medical record system platform to support a large study of tuberculosis epidemiology. *BMC Med Inform Decis Mak* [Internet]. 2012 Nov 7 [cited 2013 Jan 23];12:125. Available from:  
<http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3531253/>
  29. Guidry AF, Brinkley JF, Anderson N R T-HP. Concept Mapping to develop a framework for characterizing Electronic Data Capture (EDC) Systems. *AMIA Annual Symposium Proceedings*. 2008.
  30. Anderson N, Lee ES, Brockenbrough JS, Minie ME, Fullers S, Brinkley J, et al. Issues in Biomedical Research Data Management and Analysis: Needs and Barriers. *Journal of the American Medical Informatics Association*. 2007;14(4):478–88.
  31. Fraser HS, Blaya J. Implementing medical information systems in developing countries, what works and what doesn't. *AMIA Annu Symp Proc* [Internet]. 2010 [cited 2013 Mar 23];2010:232–6. Available from:  
<http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3041413/>
  32. Anokwa, Yaw. *Improving Clinical Decision Support in Low-Income Regions*. University of Washington; 2012.
  33. Hartung, Carl. *Open Data Kit: Technologies for Mobile Data Collection and Deployment Experiences in Developing Regions*. University of Washington; 2012.
  34. Kalogriopoulos NA, Baran J, Nimunkar AJ, Webster JG. Electronic medical record systems for developing countries: Review. 2009. p. 1730–3.
  35. Odawo P. *Improving the Use of Electronic Medical Register Systems in Kenya*. University of Washington; 2010.
  36. NASCOP. HIV Care Patient Card MOH 257.
  37. Ceusters W, Smith B, Flanagan J. Ontology and medical terminology: why description logics are not enough. *Proceedings of TEPR*. Citeseer; 2003. p. 14.
  38. Noy NF. Semantic integration: a survey of ontology-based approaches. *ACM Sigmod Record*. ACM; 2004;33(4):70.

39. OWL Web Ontology Language Reference [Internet]. [cited 2011 Oct 28]. Available from: <http://www.w3.org/TR/owl-ref/>
40. Héja G, Surján G, Varga P. Ontological analysis of SNOMED CT. BMC medical informatics and decision making [Internet]. 2008 Jan;8 Suppl 1(Suppl 1):S8. Available from: <http://www.biomedcentral.com/1472-6947/8/S1/S8>
41. Julina JKJ, Thenmozhi D. Ontology based EMR for decision making in health care using SNOMED CT. 2012. p. 514–9.
42. Zamboulis L, Poulouvassilis A, Roussos G. Flexible data integration and ontology-based data access to medical records. 2008. p. 1–6.
43. Hadzic M, Dillon T, Chang E. Use of Digital Ecosystem and Ontology Technology for Standardization of Medical Records. 2007. p. 595–601.
44. Iqbal AM, Shepherd M, Abidi SSR. An Ontology-Based Electronic Medical Record for Chronic Disease Management. 2011. p. 1–10.
45. Richesson R, Andrews J, Krischer J. Use of SNOMED CT to represent clinical research data: a semantic .... Journal of the American Medical .... 2006;
46. Das A. Epoch Ontologies for Translational Trials Design and Management. Epoch. 2008.
47. Modgil S, Hammond P. Decision support tools for clinical trial design. Artificial Intelligence In Medicine. 2003;27(2):181–200.
48. Tu SW, Carini S, Rector A, MacCallum P, Toujilov I, Harris S, et al. OCRe : An Ontology of Clinical Research. Diabetes. 2009;(1):1–20.
49. PhD IS. Trial Banks: An Informatics Foundation for Evidence-Based Medicine. 1997;
50. Li H, Gennari JH, Brinkley JF. Model Driven Laboratory Information Management Systems. AMIA Annu Symp Proc. 2006. p. 484–8.
51. Human Studyome [Internet]. [cited 2013 Jan 23]. Available from: <http://rctbank.ucsf.edu/>
52. Sim I, Rennels G. A trial bank model for the publication of clinical trials. Proc Annu Symp Comput Appl Med Care [Internet]. 1995 [cited 2013 Mar 23];863–7. Available from: <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2579216/>
53. Tierney, Willam M, Beck, Eduard J, Gardner, Reed M, Musick, Beverly, Shields, Mark, Shiyonga, Naomi M, et al. Viewpoint: A Pragmatic Approach to Constructing a Minimum Data Set for Care of Patients with HIV in Developing Countries. Journal

- of the American Medical Informatics Association [Internet]. 2006 Jun;13(3):253–60. Available from: <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1513663/>
54. Maggi F, Cycle X. A Survey of Probabilistic Record Matching Models, Techniques and Tools.
  55. Blakely T, Salmond C. Probabilistic record linkage and a method to calculate the positive predictive value. *International Journal of Epidemiology*. 2002;31(6):1246–52.
  56. Hillestad R, Bigelow JH, Chaudhry B, Dreyer P, Greenberg MD, Meili RC, et al. Identity crisis: An examination of the costs and benefits of a unique patient identifier for the US health care system. 2008.
  57. Grannis SJ, Overhage JM, Hui S, McDonald CJ. Analysis of a Probabilistic Record Linkage Technique without Human Review. *AMIA Annu Symp Proc*. 2003. p. 259–63.
  58. Zingmond DS, Ye Z, Ettner SL, Liu H. Linking hospital discharge and death records--accuracy and sources of bias. *Journal of clinical epidemiology*. 2004;57(1):21–9.
  59. DuVall SL, Kerber RA, Thomas A. Extending the Fellegi-Sunter probabilistic record linkage method for approximate field comparators. *Journal of Biomedical Informatics*. 2010;43(1):24–30.
  60. Sideli RV, Friedman C. Validating patient names in an integrated clinical information system. *Proc Annu Symp Comput Appl Med Care*. 1991. p. 588–92.
  61. Patman F, Shaefer L. Is Soundex good enough for you? On the hidden risks of Soundex-based name searching. *Language Analysis Systems, Inc, Herndon*. 2001;
  62. Stanier A. How accurate is Soundex matching. *Computers in Genealogy*. 1990;3(7):286–8.
  63. Anderson WN, Kotzé AE. Sounds like “Sutu.” *South African Journal of African Languages* [Internet]. 2005 Jan 1 [cited 2013 Sep 30];25(2):111–23. Available from: <http://www.tandfonline.com/doi/abs/10.1080/02572117.2005.10587254>
  64. Newcombe HG. *NYSIIS Algorithm Handbook of Record Linkage*. New York, NY: Oxford University Press; 1988.
  65. Philips L. Hanging on the metaphone. *Computer Language*. 1990;7(12 (December)).
  66. Hood D. Caverphone: Phonetic matching algorithm. *Technical Paper CTP060902*, University of Otago, New Zealand. 2002;

67. Louie B, Mork P, Martin-Sanchez F, Halevy A, Tarczy-Hornoch P. Data integration and genomic medicine. *Journal of Biomedical Informatics*. 2007;40(1):5–16.
68. Shaker R, Mork P, Brockenbrough JS, Donelson L, Tarczy-Hornoch P. The biomediator system as a tool for integrating biologic databases on the web. *Proceedings of the Workshop on Information Integration on the Web*. 2004.
69. Donelson L, Tarczy-Hornoch P, Mork, P, Dolan C, Mitchell JA, Barrier M, et al. The BioMediator System as a Data Integration Tool to Answer Diverse Biologic Queries. *Proceedings of the 11th World Congress on Medical Informatics*.
70. Detwiler LT, Shaw M, Brinkley JF. *Ontology View Query Management*. Annual Symposium of the American Medical Informatics Association. Washington, DC.; 2010. p. 1023.
71. Brinkley JF, Detwiler LT. A Query Integrator and Manager for the Query Web. *Journal of Biomedical Informatics [Internet]*. 2012;(0):- . Available from: <http://www.sciencedirect.com/science/article/pii/S1532046412000536>
72. Wache H, Voegelé T, Visser U, Stuckenschmidt H, Schuster G, Neumann H, et al. Ontology-based integration of information-a survey of existing approaches. *IJCAI-01 Workshop: Ontologies and Information Sharing*. Citeseer; 2001. p. 108–17.
73. Min H, Manion FJ, Goralczyk E, Wong Y-N, Ross E, Beck JR. Integration of prostate cancer clinical data using an ontology. *Journal of Biomedical Informatics*. 2009;42(6):1035–45.
74. Murphy SN, Weber G, Mendis M, Gainer V, Chueh HC, Churchill S, et al. Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2). *Journal of the American Medical Informatics Association : JAMIA*. BMJ Publishing Group Ltd; 2010 Jan 1;17(2):124–30.
75. Sioutos N, de Coronado S, Haber MW, Hartel FW, Shaiu W-L, Wright LW. NCI Thesaurus: A semantic model integrating cancer-related clinical and molecular information. *Journal of Biomedical Informatics*. 2007;30–43.
76. Mork P, Halevy A, Tarczy-Hornoch P. A model for data integration systems of biomedical data applied to online genetic databases. *Proceedings of the AMIA Symposium*. American Medical Informatics Association; 2001;473–7.
77. Odhiambo-Otieno GW. Evaluation of existing District Health Management Information Systems: A case study of the District Health Systems in Kenya. *International Journal of Medical Informatics [Internet]*. 2005 Sep;74(9):733–44. Available from: <http://www.sciencedirect.com/science/article/pii/S1386505605000560>
78. Abernethy N. Automating social network models for tuberculosis contact investigation. Stanford; 2005.

79. Abernethy NF, DeRimer K, Small PM. Methods to Identify Standard Data Elements in Clinical and Public Health Forms. AMIA Annu Symp Proc [Internet]. 2011 [cited 2013 Feb 18];2011:19–27. Available from: <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3243268/>
80. Guidry AF, Walson JL, Abernethy NF. Linking information systems for HIV care and research in Kenya. Proceedings of the 1st ACM International Health Informatics Symposium [Internet]. New York, NY, USA: ACM; 2010. p. 531–5. Available from: <http://doi.acm.org/10.1145/1882992.1883078>
81. RadLex - Summary | NCBO BioPortal [Internet]. [cited 2012 May 1]. Available from: <http://bioportal.bioontology.org/ontologies/40885>
82. Open Source Clinical Trials Software | OpenClinica | [Internet]. [cited 2012 May 1]. Available from: <https://community.openclinica.com/>
83. Hannan TJ, Rotich JK, Odero WW, Menya D, Esamai F, Einterz RM, et al. The Mosoriot medical record system: design and initial implementation of an outpatient electronic record system in rural Kenya. Int J Med Inform [Internet]. 2000 Oct [cited 2012 May 1];60(1):21–8. Available from: <http://www.ncbi.nlm.nih.gov/pubmed/10974639>
84. Hillestad R, Bigelow JH, Chaudhry B, Dreyer P, Greenberg MD, Meili RC, et al. Identity crisis: An examination of the costs and benefits of a unique patient identifier for the US health care system. The RAND Corporation; 2008.
85. NASCOP. HIV Care Patient Card MOH 257 [Internet]. Available from: [http://www.nascop.or.ke/library/3d/HIV Care Patient Card MOH 257.pdf](http://www.nascop.or.ke/library/3d/HIV%20Care%20Patient%20Card%20MOH%20257.pdf)
86. WHO Patient Monitoring Guidelines for HIV Care and Antiretroviral Therapy (ART) [Internet]. World Health Organization; Available from: <http://www.who.int/3by5/capacity/ptmonguidelinesfinalv1.PDF>
87. Patient Care Booklet [Internet]. Republic of Namibia Ministry of Health and Social Services; Available from: <http://www.aidstar-one.com/sites/default/files/Namibia%20MOHSS%20Patient%20Care%20Booklet.pdf>
88. HIV Care/ART Card [Internet]. Republic of Uganda; Available from: [http://www.iedea-ua.org/joomla/index.php?option=com\\_content&view=category&id=64:forms-a-keys&Itemid=62&layout=default](http://www.iedea-ua.org/joomla/index.php?option=com_content&view=category&id=64:forms-a-keys&Itemid=62&layout=default)
89. Patient Record Form CTC 2 [Internet]. Tanzania National Care and treatment Programme; Available from: [http://www.iedea-ua.org/joomla/index.php?option=com\\_content&view=article&id=128:tanzania-used-by-moh-ctc2-form&catid=72:forms-a-keys-tanzania&Itemid=62](http://www.iedea-ua.org/joomla/index.php?option=com_content&view=article&id=128:tanzania-used-by-moh-ctc2-form&catid=72:forms-a-keys-tanzania&Itemid=62)

90. Common Core Data Elements 2009 CSTE Position Statement 09-SI-01 [Internet]. Available from: <http://cste.org/ps2009/09-SI-01.pdf>
91. Information Technology @ Johns Hopkins-Patient Identification System [Internet]. [cited 2010 Jun 3]. Available from: <http://it.jhu.edu/fas/pid.html>
92. Welcome to the NCBO BioPortal | BioPortal [Internet]. Available from: <http://bioportal.bioontology.org/>
93. The Protégé Ontology Editor and Knowledge Acquisition System [Internet]. [cited 2012 May 2]. Available from: <http://protege.stanford.edu/>
94. Empiric Therapy of Helminth Co-infection to Reduce HIV-1 Disease Progression - Full Text View - ClinicalTrials.gov [Internet]. [cited 2012 May 8]. Available from: <http://clinicaltrials.gov/ct2/show/NCT00507221>
95. Struts 2 - Welcome [Internet]. [cited 2012 May 16]. Available from: <http://struts.apache.org/2.3.1.2/index.html>
96. Apache Jena - ARQ - A SPARQL Processor for Jena [Internet]. [cited 2012 May 16]. Available from: <http://jena.apache.org/documentation/query/>
97. The SAXON XSLT and XQuery Processor [Internet]. [cited 2012 May 16]. Available from: <http://saxon.sourceforge.net/>
98. Barrasa J, Corcho Ó, Gómez-pérez A. R 2 O , an Extensible and Semantically Based Database- to-ontology Mapping Language. Second Workshop on Semantic Web and Databases. Toronto, Canada; 2004. p. 1069–70.
99. Ghawi R, Cullot N. Database-to-Ontology Mapping Generation for Semantic Interoperability. Knowledge Creation Diffusion Utilization. 2007;
100. Patient Matching Module - Documentation - OpenMRS Wiki [Internet]. 2011 [cited 2013 Feb 1]. Available from: <https://wiki.openmrs.org/display/docs/Patient+Matching+Module>
101. Levenshtein VI. Binary codes capable of correcting deletions, insertions, and reversals. Soviet physics doklady. 1966. p. 707–10.
102. Winkler WE. Preprocessing of Lists and String Comparison. Record Linkage Techniques. 1985;
103. Dempster AP, Laird NM, Rubin DB. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society Series B (Methodological). 1977;1–38.
104. Fellegi IP, Sunter AB. A theory for record linkage. Journal of the American Statistical Association. 1969;1183–210.

105. SimMetrics [Internet]. SourceForge. [cited 2013 Jan 22]. Available from: <http://sourceforge.net/projects/simmetrics/>
  106. duke - Fast deduplication engine - Google Project Hosting [Internet]. [cited 2013 Jan 22]. Available from: <http://code.google.com/p/duke/>
  107. Top boys' names 2011 [Internet]. [cited 2012 Nov 30]. Available from: <http://www.babycentre.co.uk/pregnancy/naming/baby-names-2011/babycentre-top-boys-names-2011/>
  108. Top girls' names 2011 [Internet]. [cited 2012 Nov 30]. Available from: <http://www.babycentre.co.uk/pregnancy/naming/baby-names-2011/babycentre-top-girls-names-2011/>
  109. Download OpenMRS » OpenMRS [Internet]. [cited 2012 Nov 30]. Available from: <http://openmrs.org/download/>
  110. Downloads | eHealth Kenya [Internet]. [cited 2012 Nov 30]. Available from: <http://www.ehealth.or.ke/facilities/downloads.aspx>
  111. Jansen BJ, Spink A, Saracevic T. Real life, real users, and real needs: a study and analysis of user queries on the web. *Information Processing & Management* [Internet]. 2000 Mar 1;36(2):207–27. Available from: <http://www.sciencedirect.com/science/article/pii/S0306457399000564>
  112. Markatos E. On caching search engine query results. *Computer Communications* [Internet]. 2001 Feb 1;24(2):137–43. Available from: <http://www.sciencedirect.com/science/article/pii/S014036640000308X>
  113. Spink A, Wolfram D, Jansen MJB, Saracevic T. Searching the web: The public and their queries. *Journal of the American Society for Information Science and Technology* [Internet]. 2001;52(3):226–34. Available from: [http://dx.doi.org/10.1002/1097-4571\(2000\)9999:9999<::AID-ASI1591>3.0.CO;2-R](http://dx.doi.org/10.1002/1097-4571(2000)9999:9999<::AID-ASI1591>3.0.CO;2-R)
  114. CJ van Rijsbergen. *Information Retrieval*. 2nd ed. London: Butterworths; 1979.
  115. Hoxmeier JA, DiCesare C. System response time and user satisfaction: An experimental study of browser-based applications. *Proceedings of the Association of Information Systems Americas Conference*. Citeseer; 2000. p. 140–5.
  116. Galletta DF, Henry, Raymond, McCoy, Scott, Polak, Peter. *Web site delays: How tolerant are users?* Barcelona; 2002.
  117. *Health Information Technology for Economic and Clinical Health Act*. 111-115 Feb 17, 2009.
-

**Appendix A. NASCOP Patient Comprehensive Care Card –  
MOH 257; “Blue Card”**

Facility Name..... Patient Clinic Number.....  
PATIENT PROFILE

**FOR SWITCH TO 2ND LINE  
REGIMEN ONLY**  
8. Clinical treatment failure  
9. Immunologic failure  
10. Virologic failure

*Version: October 2010*



Clinician Initials

(ah)

FS = Food Support Infant Feeding Practices = EBF, ERF, MF if <2yrs)			

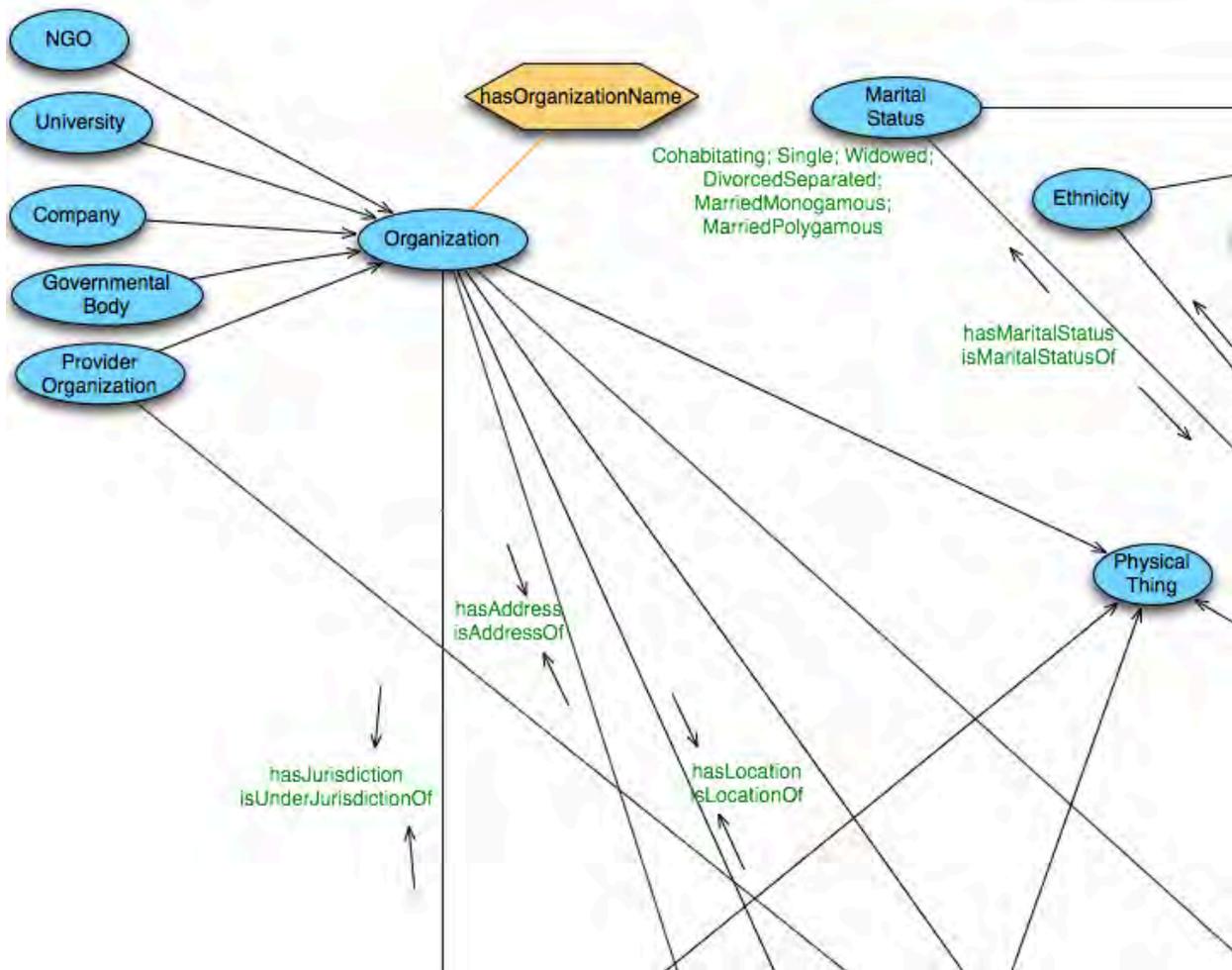
Follow-up Education Support and Preparation for ARV Therapy				
	Date/comments	Date/comments	Date/comments	Date/comments
<b>Educate on basics, prevention, disclosure</b>	<b>Basic HIV education, transmission</b>			
	Prevention: abstinence, safer sex, condoms			
	Prevention: household precautions, what is safe			
	Post-test counselling: implications of results			
	Positive living			
	Testing partners			
	Disclosure: to whom disclosed (if so)			
	Family/living situation			
	Shared confidentiality			
	Reproductive choices, prevention of MTCT			
<b>Progression, Rx</b>	<b>Chills blood test:</b>			
	Progression of disease			
	<b>Available treatments/prophylaxis</b>			
	CTX, INH prophylaxis			
	Malaria prevention, IPT, ITN			
	Follow-up appointments, clinical team			
	<b>ART -- educate on essentials (locally adapted):</b>			
	Why complete adherence needed			
	<b>Adherence preparation, indicate visits</b>			
	Indicate when READY for ART: DATE/suit clinical team discussion			
<b>ART preparation, initiation, support, monitor, Rx</b>	<b>Explain dose, when to take</b>			
	What can occur, how to manage side effects			
	What to do if one forgets dose			
	What to do when travelling			
	<b>Adherence plan (schedule, aids, explain diary)</b>			
	Treatment supporter preparation			
	<b>Which doses, why missed</b>			
	ARV support group			
	How to contact clinic			
	Symptom management/palliative care at home			
<b>Home-based care, support</b>	<b>Caregiver booklet:</b>			
	Home-based care – specify			
	<b>Support groups</b>			
	Community support			

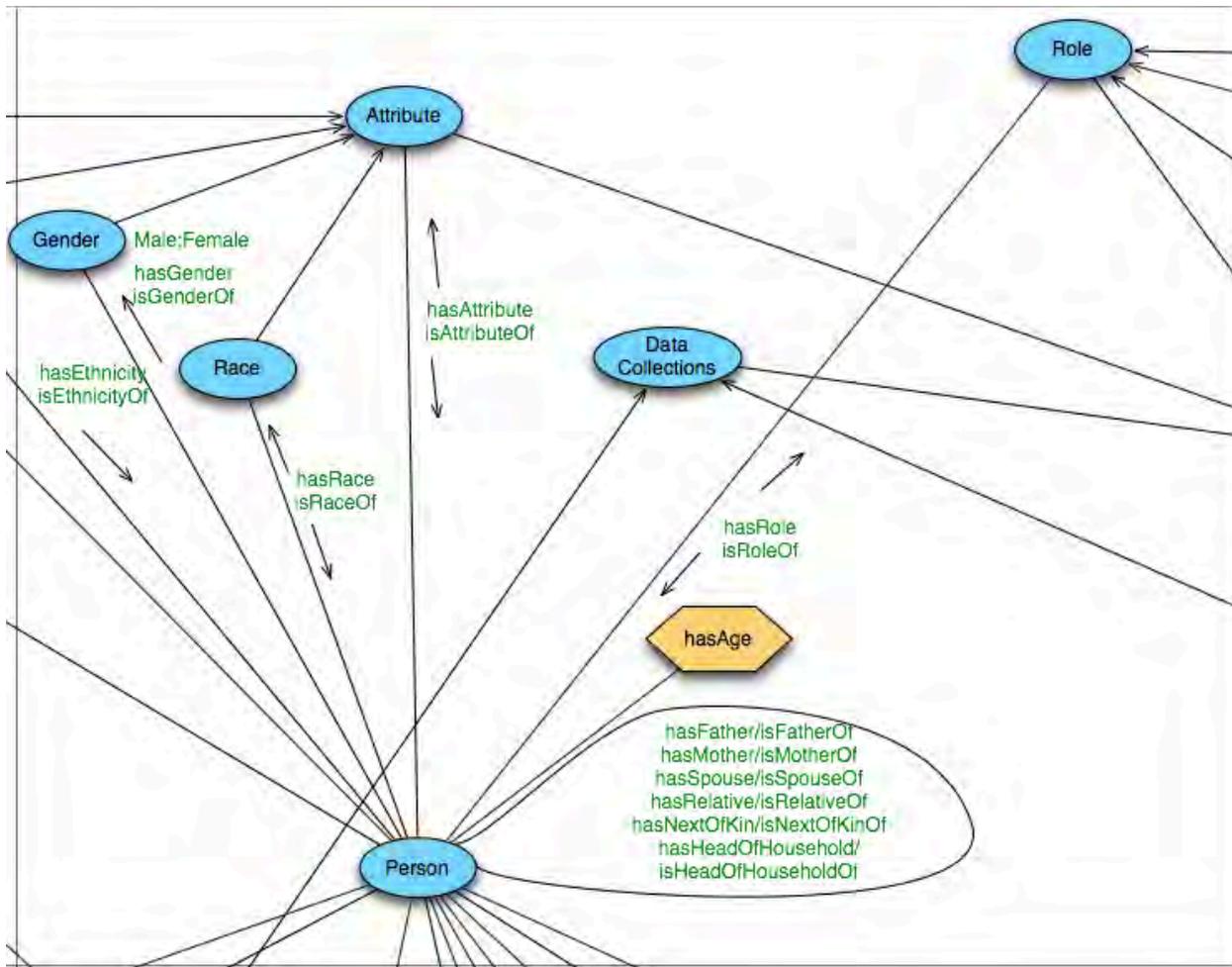
## Appendix B. Patient Identifier Ontology Data Tables

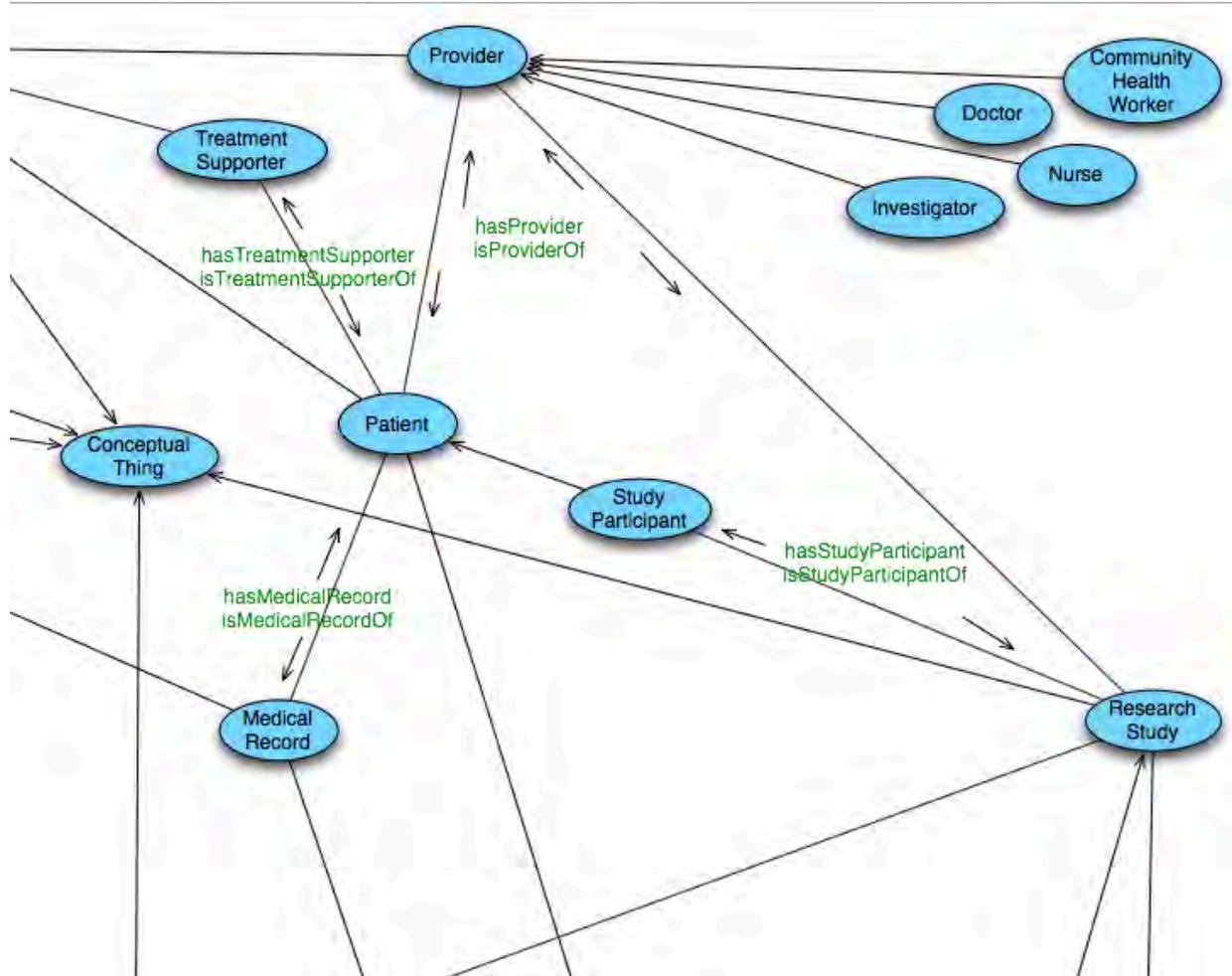
		RadLex	OpenMRS	OpenClinica	WHO - Minimum HIV/AIDS DataSet	MMRS	NASCOP	CDC-CSTE	Johns Hopkins	RAND Health	Uganda - HIV Care/ART Card	EURO - HIV care/ART Card	SERO - HIV care/ART	Namibia - HIV Care/ART Card	Tanzania - HIV Care/ART Card
1	Medical Record Number	x	x												
2	StudyID			x											
3	Patient Age	x			x		x	x			x		x	x	x
4	Patient Date of Birth	x	x	x	x	x	x	x	x	x	x	x	x	x	x
5	Patient Ethnicity	x						x							
6	Patient Gender	x	x	x	x		x	x	x		x	x	x	x	x
7	Patient First Name		x		x	x		x	x	x					
8	Patient Middle Name		x			x		x	x	x					
9	Patient Last Name				x	x		x	x	x					
10	Patient's Mother's First name					x									
11	Patient's Home Village		x			x									
12	Patient Name	x	x				x				x	x	x	x	x
13	ClinicID				x						x	x	x	x	x
14	Unique Patient Identifier				x		x				x	x	x	x	x
15	Street Address		x		x			x				x			x
16	City		x		x			x				x			
17	State/territory		x		x			x				x			
18	Zip Code		x		x			x		x		x			
19	Country		x					x				x			
20	Telephone				x		x	x	x		x	x	x	x	x
21	Race							x	x						

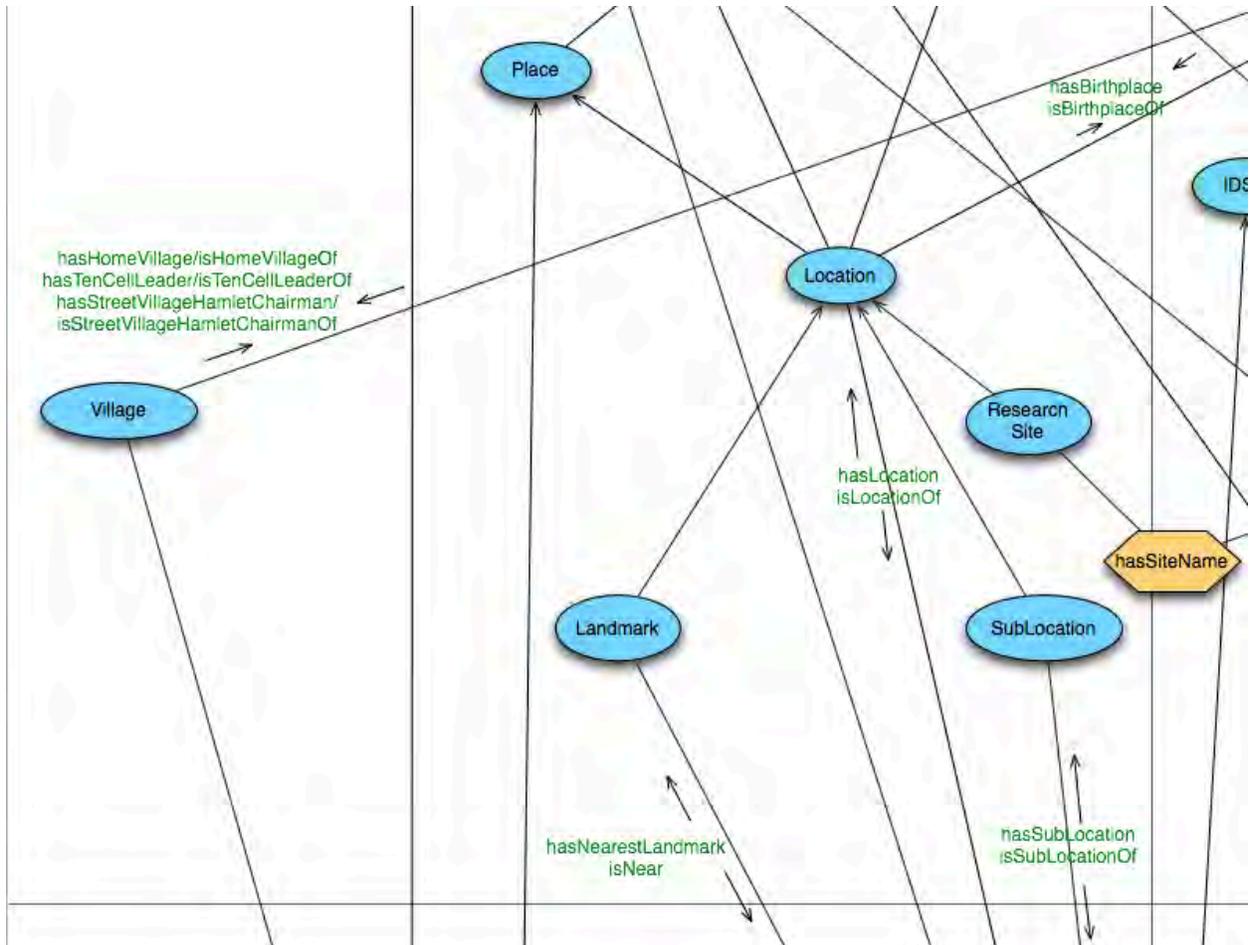


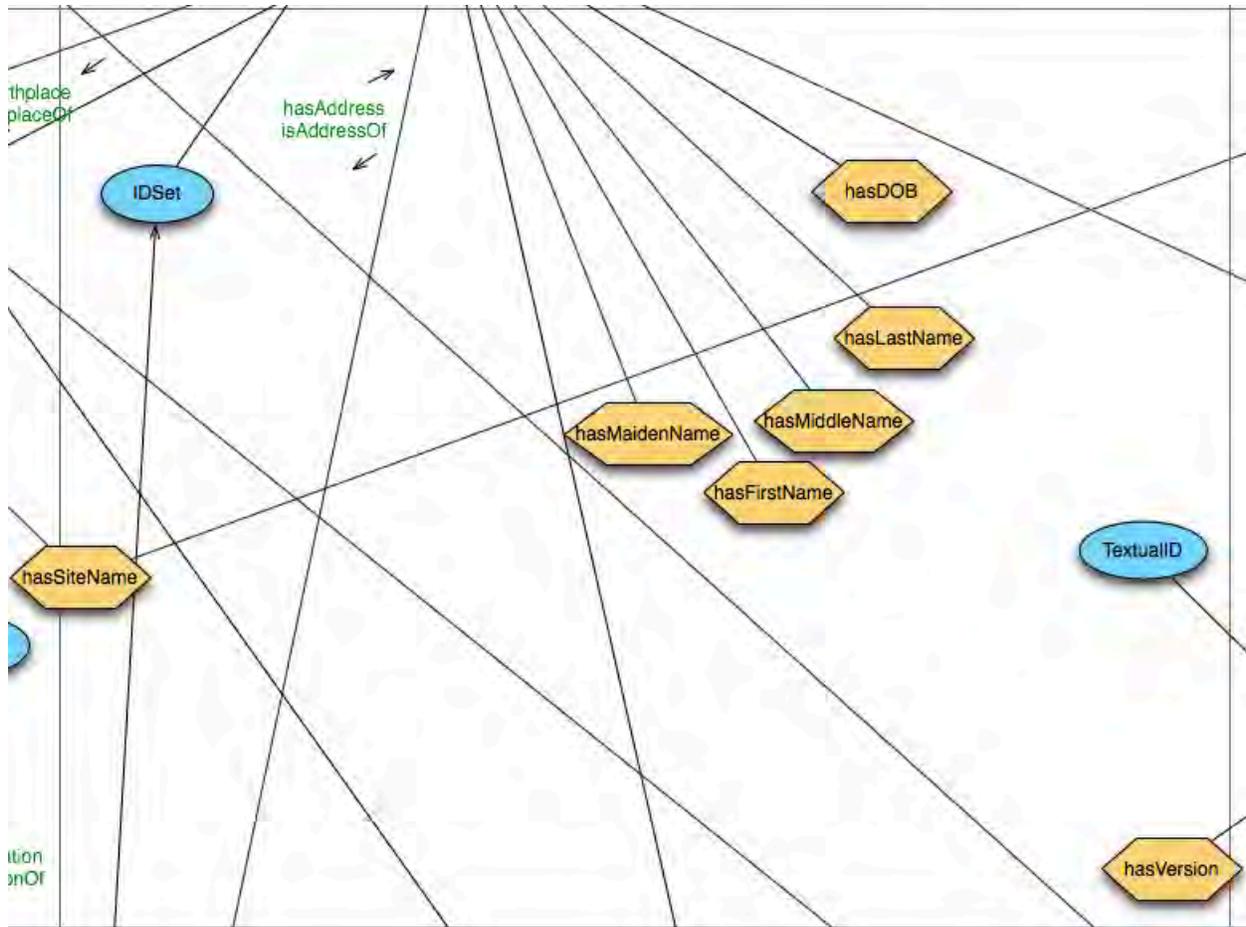


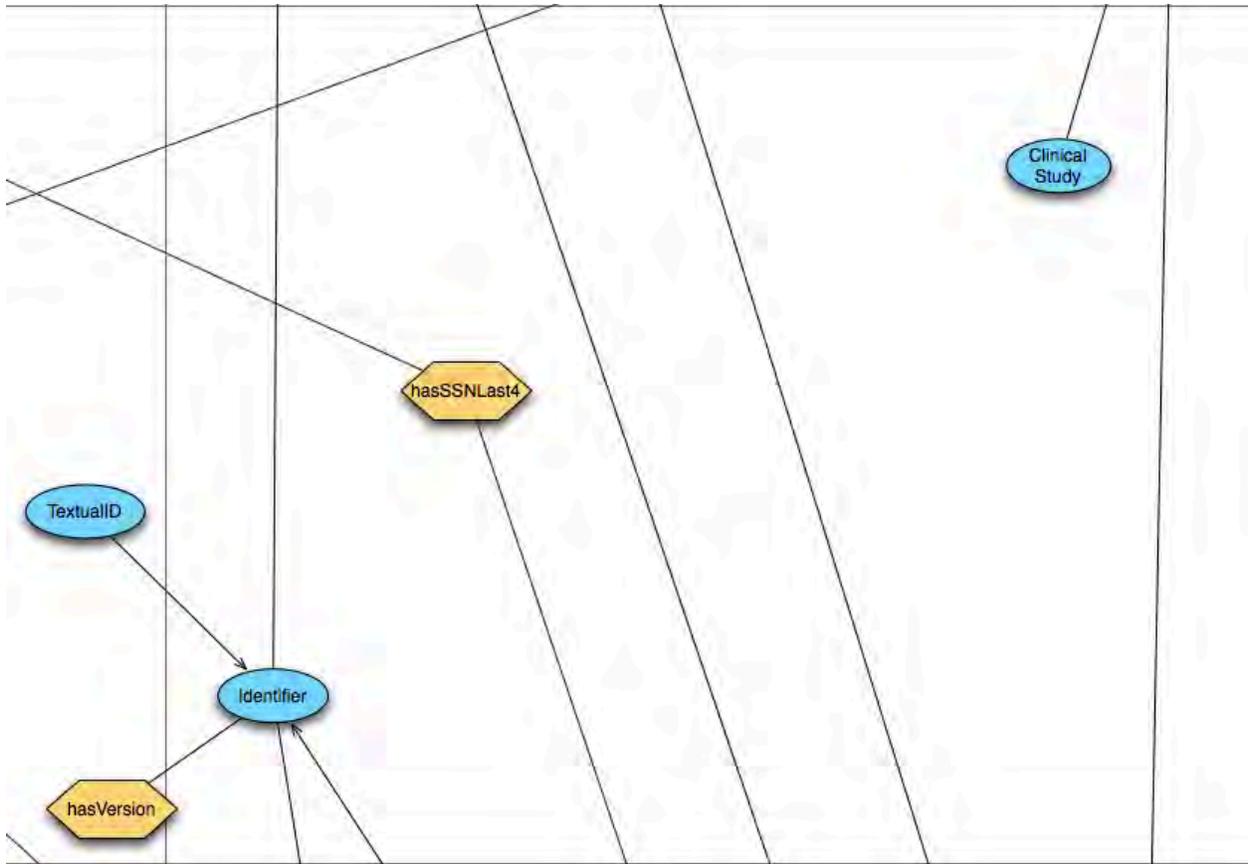


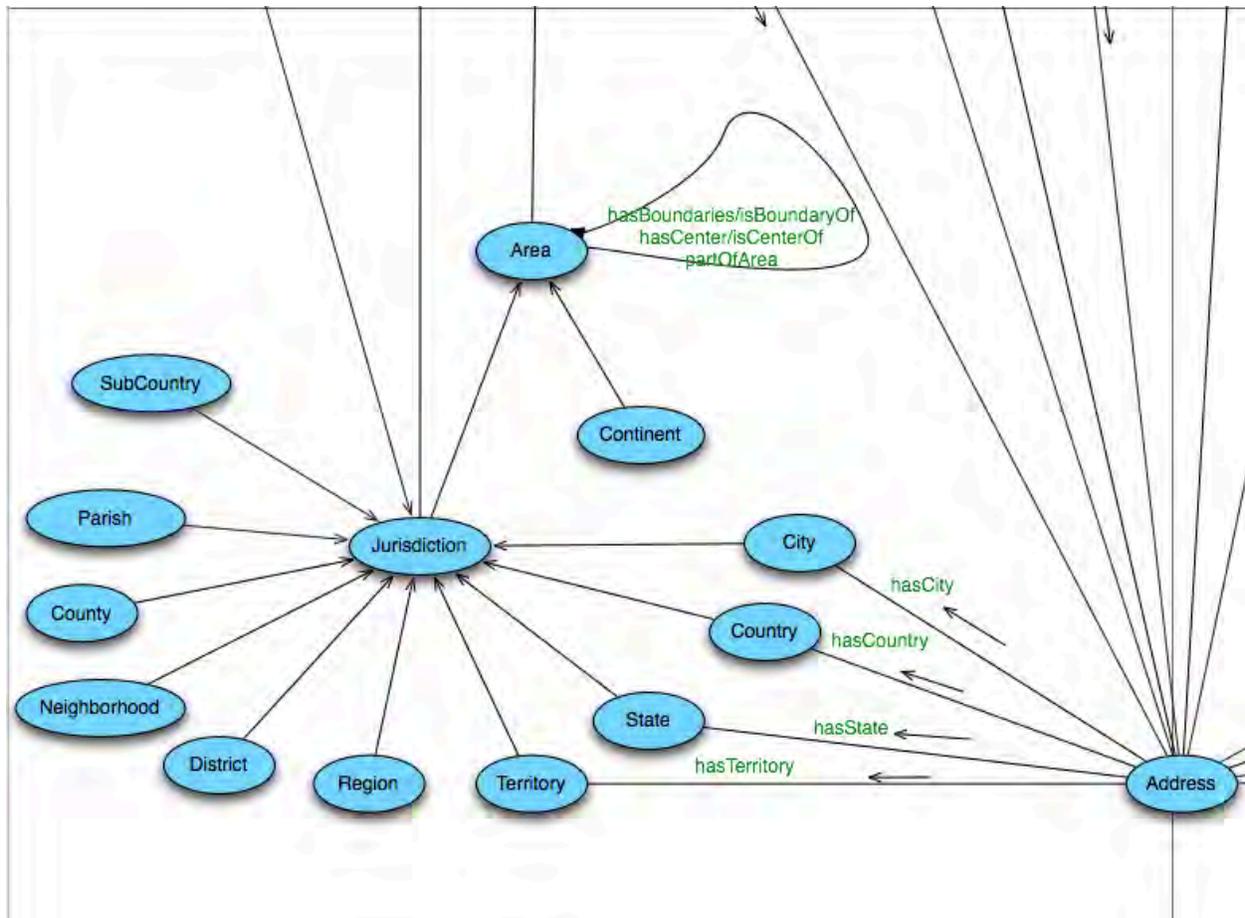


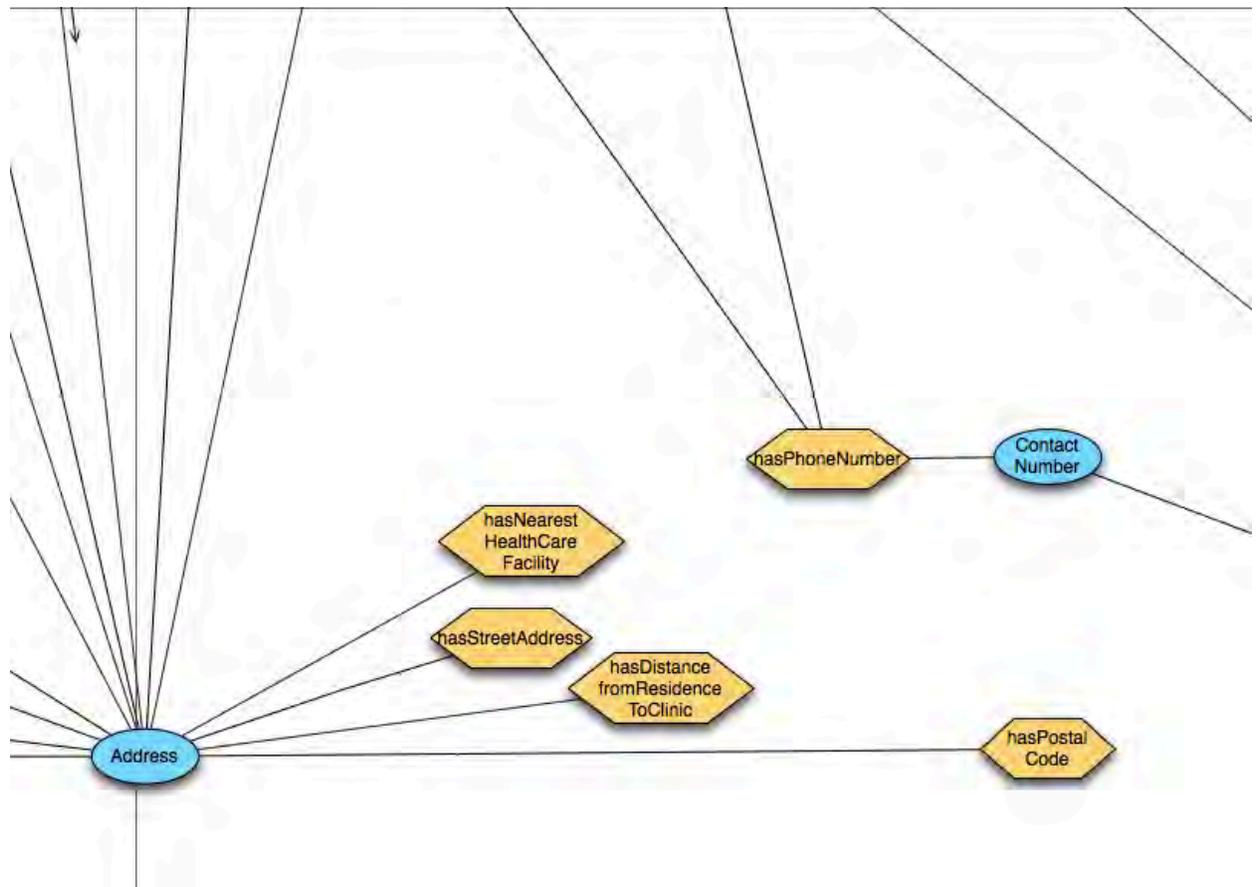


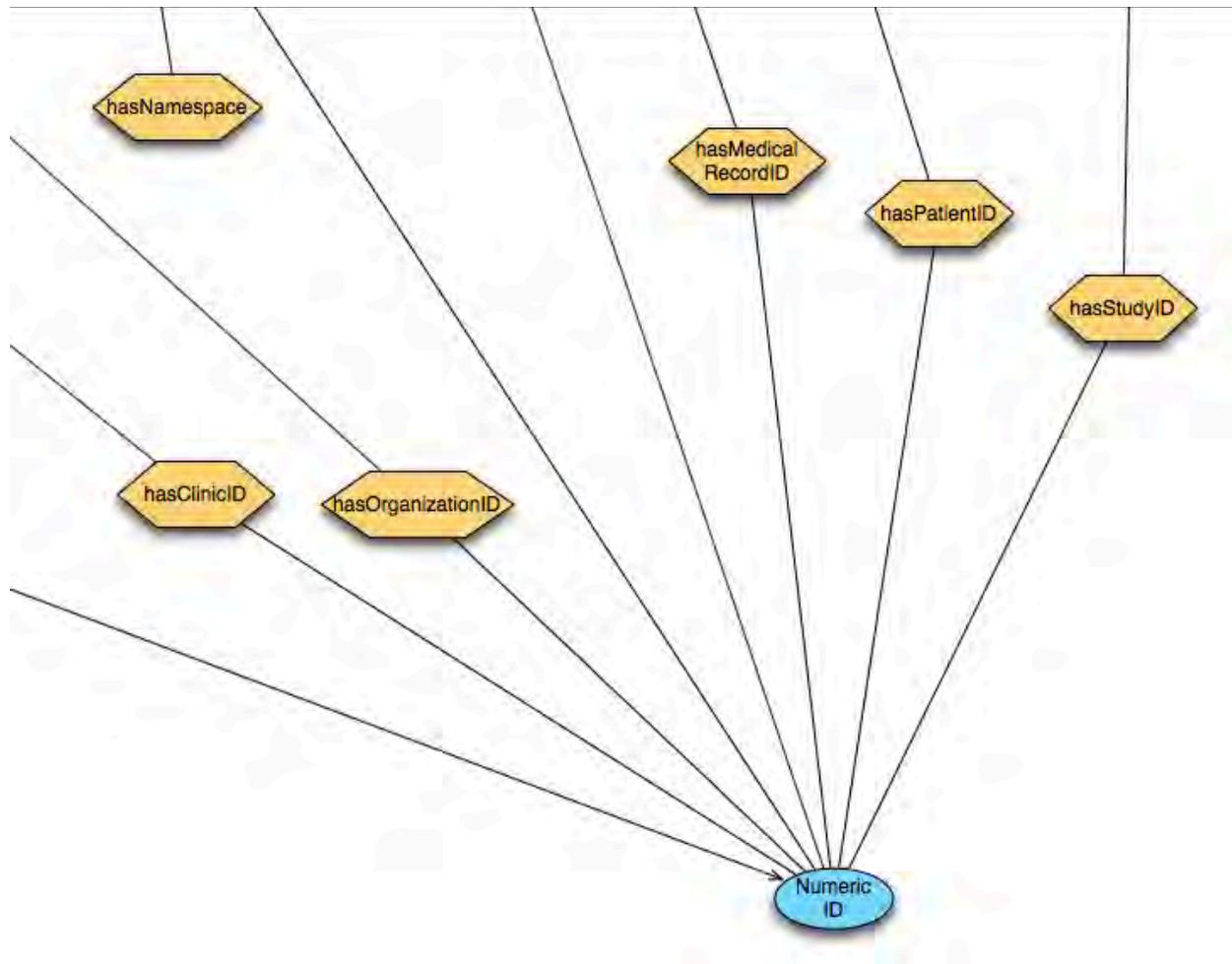












## Appendix D. HIV Opportunistic Infections

Aspergillosis  
Bacterial Respiratory Disease  
Bacterial Enteric Infections  
Bartonellosis  
Chagas Disease  
Coccidioidomycosis  
Cryptococcosis  
Cryptosporidiosis  
Cytomegalovirus Disease  
Disseminated *Mycobacterium avium* Complex Disease  
Hepatitis B Virus Infection  
Hepatitis C Virus Infection  
Herpes Simplex Virus Disease  
HHV-6 and HHV-7 Disease  
Histoplasmosis  
Human Herpesvirus-8 Disease  
Human Papillomavirus Disease  
Isosporiasis  
Leishmaniasis  
Malaria  
Microsporidiosis  
Mucocutaneous Candidiasis  
*Mycobacterium tuberculosis* Infection and Disease  
*Penicilliosis marneffe*  
*Pneumocystis* Pneumonia  
Progressive Multifocal Leukoencephalopathy/JC Virus Infection  
Syphilis  
*Toxoplasma gondii* Encephalitis  
Varicella-Zoster Virus Diseases

## Appendix E. WHO Clinical Staging Tables

### WHO Clinical Staging for Adults and Adolescents

<b>Clinical stage 1</b>
<ol style="list-style-type: none"> <li>1. Asymptomatic</li> <li>2. Persistent generalized lymphadenopathy (PGL)</li> </ol>
<b>Clinical stage 2</b>
<ol style="list-style-type: none"> <li>1. Moderate unexplained weight loss (&lt;10% of presumed or measured body weight)</li> <li>2. Minor mucocutaneous manifestations (seborrheic dermatitis, popular pruritic eruptions, fungal nail infections, recurrent oral ulcerations, angular cheilitis)</li> <li>3. Herpes zoster</li> <li>4. Recurrent upper respiratory tract infections (sinusitis, tonsillitis, bronchitis, otitis media, pharyngitis)</li> </ol>
<b>Clinical stage 3</b>
<ol style="list-style-type: none"> <li>1. Unexplained severe weight loss (over 10% of presumed or measured body weight)</li> <li>2. Unexplained chronic diarrhoea for longer than one month</li> <li>3. Unexplained persistent fever (intermittent or constant for longer than one month)</li> <li>4. Persistent oral candidiasis</li> <li>5. Oral hairy leukoplakia</li> <li>6. Pulmonary tuberculosis</li> <li>7. Severe bacterial infections (e.g. pneumonia, empyema, pyomyositis, bone or joint infection, meningitis, bacteraemia)</li> <li>8. Acute necrotizing ulcerative stomatitis, gingivitis or periodontitis</li> <li>9. Unexplained anaemia (below 8 g/dl), neutropenia (below 0.5 x 10<sup>9</sup>/l) and/or chronic thrombocytopenia (below 50 x 10<sup>9</sup>/l)</li> </ol>
<b>Clinical stage 4</b>
<p><i>Conditions where a presumptive diagnosis can be made using clinical signs or simple investigations:</i></p> <ol style="list-style-type: none"> <li>1. HIV wasting syndrome</li> <li>2. Pneumocystis jiroveci pneumonia (PCP)</li> <li>3. Recurrent severe bacterial pneumonia (≥ 2 episodes within 1 year)</li> <li>4. Cryptococcal meningitis</li> <li>5. Toxoplasmosis of the brain</li> <li>6. Chronic orolabial, genital or ano-rectal herpes simplex infection for &gt;1 month</li> <li>7. Kaposi sarcoma (KS)</li> <li>8. HIV encephalopathy</li> <li>9. Extra pulmonary tuberculosis (EPTB)</li> </ol> <p><i>Conditions where confirmatory diagnostic testing is necessary:</i></p> <ol style="list-style-type: none"> <li>1. Cryptosporidiosis, with diarrhoea &gt;1 month</li> <li>2. Isosporiasis</li> <li>3. Cryptococcosis (extra pulmonary)</li> <li>4. Disseminated non-tuberculous mycobacterial infection</li> <li>5. Cytomegalovirus (CMV) retinitis or infection of the organs (other than liver, spleen, or lymph nodes)</li> <li>6. Progressive multifocal leucoencephalopathy (PML)</li> <li>7. Any disseminated mycosis (e.g. histoplasmosis, coccidiomycosis)</li> <li>8. Candidiasis of the oesophagus or airways</li> <li>9. Non-typhoid salmonella (NTS) septicaemia</li> <li>10. Lymphoma cerebral or B cell Non Hodgkin's Lymphoma</li> <li>11. Invasive cervical cancer</li> <li>10. Visceral leishmaniasis</li> <li>11. Symptomatic HIV-associated nephropathy or HIV-associated cardiomyopathy</li> </ol>

## WHO Clinical Staging for Adolescents

**Clinical stage 1**

1. Asymptomatic
2. Persistent generalized lymphadenopathy (PGL)

**Clinical stage 2**

1. Unexplained persistent hepatosplenomegaly
2. Papular pruritic eruptions
3. Extensive wart virus infection
4. Extensive molluscum contagiosum
5. Recurrent oral ulcerations
6. Unexplained persistent parotid enlargement
7. Lineal gingival erythema
8. Herpes zoster
9. Recurrent or chronic upper respiratory tract infections (otitis media, otorrhoea, sinusitis, tonsillitis)
10. Fungal nail infections

**Clinical stage 3**

1. Unexplained moderate malnutrition not adequately responding to standard therapy
2. Unexplained persistent diarrhoea (14 days or more)
3. Unexplained persistent fever (above 37.5°C, intermittent or constant, for longer than one month)
4. Persistent oral Candidiasis (after first 6 weeks of life)
5. Oral hairy leukoplakia
6. Acute necrotizing ulcerative gingivitis/periodontitis
7. Lymph node TB
8. Pulmonary TB
9. Severe recurrent bacterial pneumonia
10. Symptomatic lymphoid interstitial pneumonitis
11. Chronic HIV-associated lung disease including bronchiectasis
12. Unexplained anaemia (<8.0 g/dl), neutropenia (<0.5x10<sup>9</sup>/L3) or chronic thrombocytopenia (<50 x 10<sup>9</sup>/L3)

**Clinical stage 4**

1. Unexplained severe wasting, stunting or severe malnutrition not responding to standard therapy
2. Pneumocystis pneumonia
3. Recurrent severe bacterial infections (e.g. empyema, pyomyositis, bone or joint infection, meningitis, but excluding pneumonia)
4. Chronic herpes simplex infection; (orolabial or cutaneous of more than one month's duration, or visceral at any site)
5. Extrapulmonary TB
6. Kaposi sarcoma
7. Oesophageal candidiasis (or candidiasis of trachea, bronchi or lungs)
8. Central nervous system toxoplasmosis (after the neonatal period)
9. HIV encephalopathy
10. Cytomegalovirus (CMV) infection; retinitis or CMV infection affecting another organ, with onset at age more than 1 month
11. Extrapulmonary cryptococcosis including meningitis
12. Disseminated endemic mycosis (extrapulmonary histoplasmosis, coccidioidomycosis, penicilliosis)
13. Chronic cryptosporidiosis (with diarrhoea)
14. Chronic isosporiasis
15. Disseminated non-tuberculous mycobacterial infection
16. Cerebral or B cell non-Hodgkin lymphoma
17. Progressive multifocal leukoencephalopathy
18. HIV-associated cardiomyopathy or nephropathy

## Appendix F. HIV Ontology Data Tables

		Uganda - HIV Care/ART Card	Kenya/NASCOP - Comprehensive Care Clinic Patient Card	WHO - Minimum Data Set	EURO - HIV care/ART Card	SERO - HIV care/ART	Namibia - HIV Care/ART Card	Tanzania - HIV Care/ART Card
1	Last Name	x	x	x	x	x	x	x
2	First Name	x	x	x	x	x	x	x
3	Sex	x	x	x	x	x	x	x
4	Date of Birth	x	x	x	x	x	x	x
5	Age at registration for HIV Care	x	x	x		x	x	x
6	Marital Status	x	x	x	x	x	x	x
7	Unique ID Number	x	x	x	x	x	x	x
8	Patient clinic ID number	x	x	x	x	x	x	x
9	Patient Address			x	x		x	x
10	Distance from residence to clinic					x		
11	Patient postal address		x			x		
12	Sub-county	x						
13	Parish	x						
14	District	x (LC1)	x			x		x
15	Location		x					
16	Sub-location		x					
17	Landmark		x					
18	Nearest Health Centre		x					
19	Street/Village							x
20	Street/Village/Hamlet Chairman							x
21	Ten Cell Leader							x
22	Head of household							x
23	Telephone	x	x	x	x	x	x	x
24	Positive HIV test confirmed			x				
25	HIV Subtype	x		x	x		x	
26	Date positive HIV test confirmed	x	x	x	x	x	x	x
27	Site where HIV test confirmed	x	x	x	x	x		
28	Entry point into HIV care	x	x	x	x	x	x	x
29	City where facility is located providing HIV Care currently					x		

30	State Province where facility is located providing HIV care currently					x		
31	District where facility is located providing HIV care currently	x		x	x	x		x
32	Health unit-facility where HIV care currently received	x	x	x	x	x		x
33	District clinician/team	x		x	x			
34	Name of treatment supporter	x	x	x	x	x	x	x
35	Address of treatment supporter	x	x	x	x	x	x	
36	Names of children/partners/family members	x	x	x	x	x	x	x
37	child/partner/family relation							x
38	Child/partner/family member gender					x		
39	Child/partner/family member HIV status	x	x	x	x	x	x	x
40	Child/partner/family member HIV status date				x			
41	Child/partner/family member HIV care status	x	x			x	x	x
42	Child/partner/family member unique ID	x	x	x	x	x	x	x
43	Child/partner/family age	x	x		x	x	x	x
44	Child/partner/family member age or date of birth at enrollment			x				
45	Drug allergies	x	x	x			x	x
46	Antiretroviral treatment prior to enrolment	x	x	x	x	x	x	x
47	Date determined medically eligible to start ART	x	x	x	x	x	x	x
48	Why medically eligibility to start ARV therapy	x	x	x	x		x	x
49	WHO clinical stage when medically eligible	x	x	x	x	x	x	x
50	CD4 count or % or TLC count if medically eligible based on CD4 or TLC	x	x	x	x	x	x	x
51	Date determined medically eligible and ready to start ART (prepared for adherence)	x		x	x		x	x
52	Date medically eligible, ready AND selected to begin ART at the facility	x		x			x	x
53	Date transferred in from another treatment facility on ART	x	x	x	x	x	x	x
54	Location transferred from	x	x	x	x	x	x	
55	Date ART started at original clinic	x	x	x	x	x	x	x
56	ART cohort (start-up group)	x	x	x	x		x	
57	Clinical stage at start of ART	x	x			x	x	x
58	Functional status at start of ART	x		x			x	
59	Body weight at start of ART	x	x	x		x	x	
60	Height at start of ART (for children)			x		x		
61	Height		x					
62	BMI (Adults)		x					
63	First ARV regimen at this facility	x	x	x	x	x	x	x
64	Substitute ARVs within first-line regimen (first instance) Date	x	x	x	x	x	x	x
65	Reason for substitution within first-line regimen	x	x	x	x	x	x	x
66	ARV regimen after first substitution	x	x	x	x	x	x	x

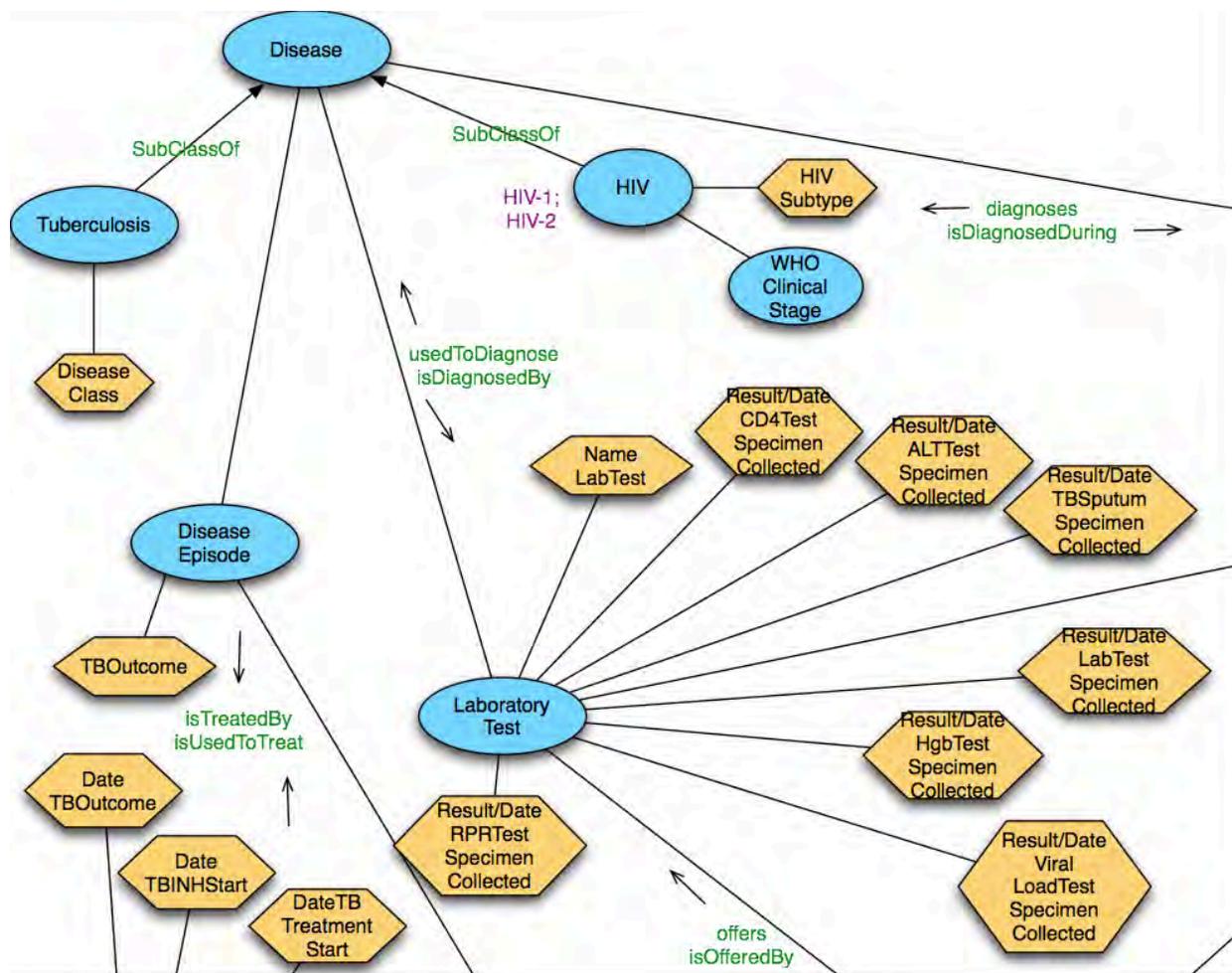
67	Substitute ARVs within first-line regimen (second instance) Date	x	x	x	x	x	x	x
68	New first-line ARV regimen following second substitution	x	x	x	x	x	x	x
69	Reason for second substitution	x	x	x	x	x	x	x
70	Switch to second-line ARV regimen Date	x	x	x	x	x	x	x
71	Reason for switch to second-line regimen or substitution within second-line regimen	x	x	x	x	x	x	x
72	Second-line ARV regimen (first switch)	x	x	x	x	x	x	x
73	Repeat switch or substitution within second-line regimens - as many times as needed		x	x	x	x	x	x
74	Stopped or Lost - ART treatment interruptions	x			x	x	x	x
75	When ART interrupted first instance, stopped or lost	x	x	x	x	x	x	x
76	If stopped ART first instance, reason	x	x	x	x	x	x	x
77	Date ART restarted first instance	x	x	x	x	x	x	x
78	Date ART interrupted second instance, stopped or lost	x	x	x	x	x	x	x
79	If stopped second instance, reason	x	x	x	x	x	x	x
80	Date ART restarted second instance	x	x	x	x	x	x	x
81	Date transferred out with records	x	x	x	x	x	x	
82	Location transferred to	x	x	x	x	x	x	
83	If temporarily LOST, indicate date	x	x	x	x	x		
84	If dropped, indicate date	x	x	x	x	x	x	
85	Date of death	x	x	x	x	x	x	
86	TB treatment in last 2 years, end date						x	
87	Outpatient encounter date	x	x	x		x	x	x
88	Visit type	x	x	x			x	
89	Next scheduled outpatient visit date	x	x	x		x	x	x
90	Duration in months since first starting ART	x	x				x	
91	Months on current ARV regimen	x	x	x			x	
92	Functional status	x		x		x	x	x
93	WHO clinical stage	x	x			x	x	x
94	Body weight	x	x	x		x	x	x
95	Blood Pressure		x					
96	Height (for children)	x		x		x		
97	Height		x					
98	BMI		x					
99	TB status	x	x	x			x	x
100	TB TBRx #		x	x		x		
101	TB treatment or INH start/stop date			x		x		
102	TB Disease class					x		
103	TB Regimen					x		
104	TB Registration District/Health Centre					x		
105	TB Outcome/Date					x		
106	Pregnancy Status		x	x		x		x
107	Pregnant EDD	x	x	x			x	x
108	Pregnancy/family planning in women of	x	x	x			x	

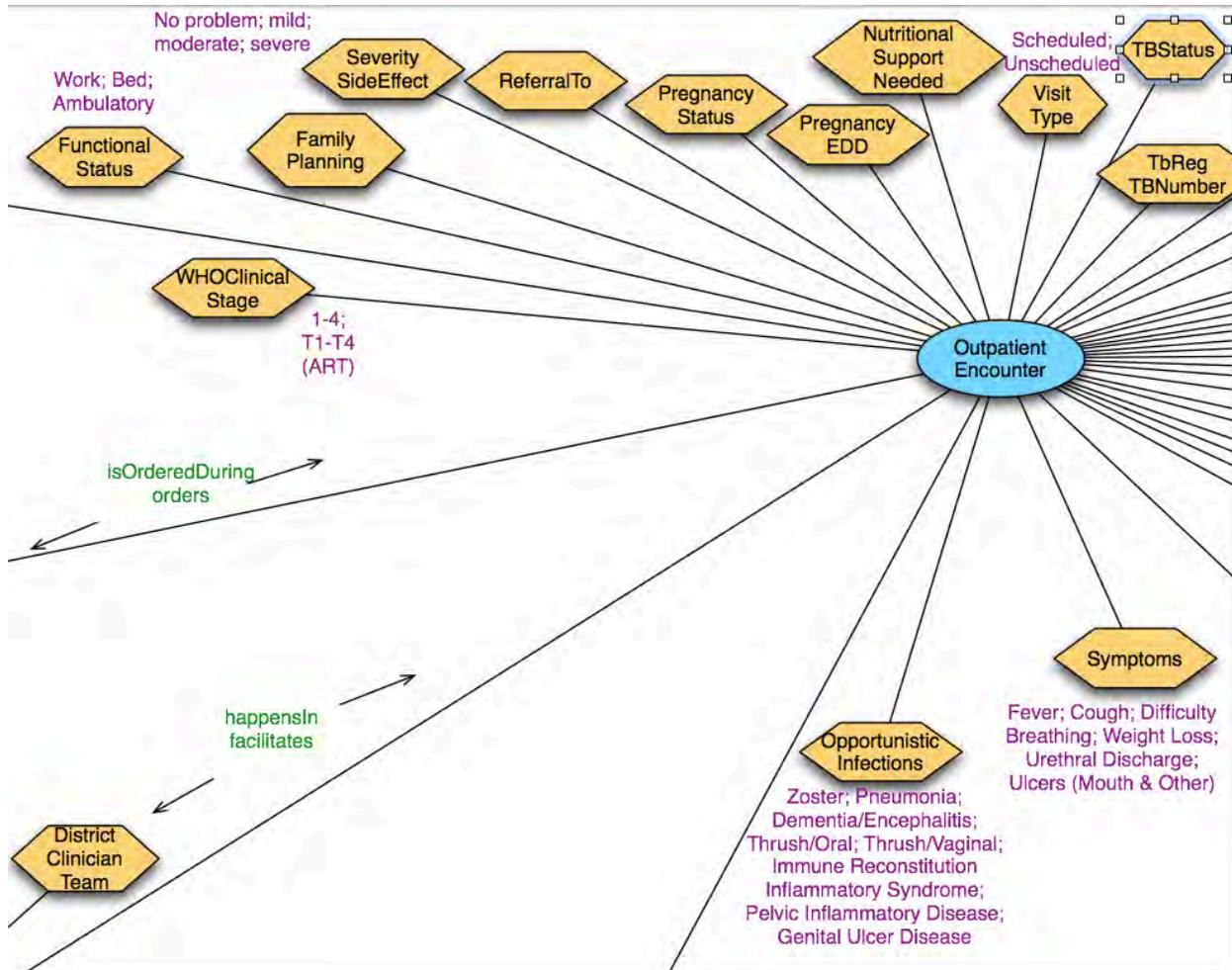
	childbearing age							
109	Family planning method(s)	x	x	x		x	x	
110	Refer for or link with other clinical care, PMTCT, supportive care	x	x	x		x	x	
111	Potential medication side-effects or other problems	x	x	x		x	x	x
112	Severity of side-effect(s)			x				
113	New symptoms/diagnoses/opportunistic infections	x	x	x		x	x	x
114	Prophylaxis medication name, dose and start date	x (name, dose)			x	x (name)		
115	Prophylaxis medications stop date			x				
116	Adherence to Cotrimoxazole	x	x	x			x	x
117	Cotrimoxazole dispensed		x				x	x
118	INH dispensed		x					
119	Reason for discontinuation of prophylaxis medication			x				
120	Other medications dispensed		x				x	x
121	Antiretroviral drug name, dose	x	x			x	x	x
122	Antiretroviral medication interruption and restart dates listed in the ART section		x				x	x
123	ARV adherence assessment	x	x			x	x	x
124	Reason for missing ARV doses/adherence problems	x	x	x			x	x
125	Laboratory test dates and names	x (name)	x	x		x (name)	x	x
126	CD4 (# or %)		x	x			x	x
127	Hgb		x	x			x	x
128	RPR		x					
129	TB Sputum		x	x				
130	ALT						x	x
131	Viral Load						x	
132	Number of hospital days since last outpatient visit	x	x				x	x
133	TB Status - Start 1st line initial regimen				x			
134	Hep B - Start 1st line initial regimen				x			
135	Hep C - Start 1st line initial regimen				x			
136	IDU Status				x	x		
137	IDU Substitution therapy					x		
138	At Risk Population		x					
139	PPW Services - Disclosure		x					
140	PPW Services - Partner tested		x					
141	PPW Services - Condoms		x			x		
142	PPW Services - Screened STI		x					
143	Home-based care provided by	x					x	
144	Treatment Supporter Relationship		x					
145	Treatment Supporter Telephone Number	x	x				x	x
146	District		x					
147	Location		x					

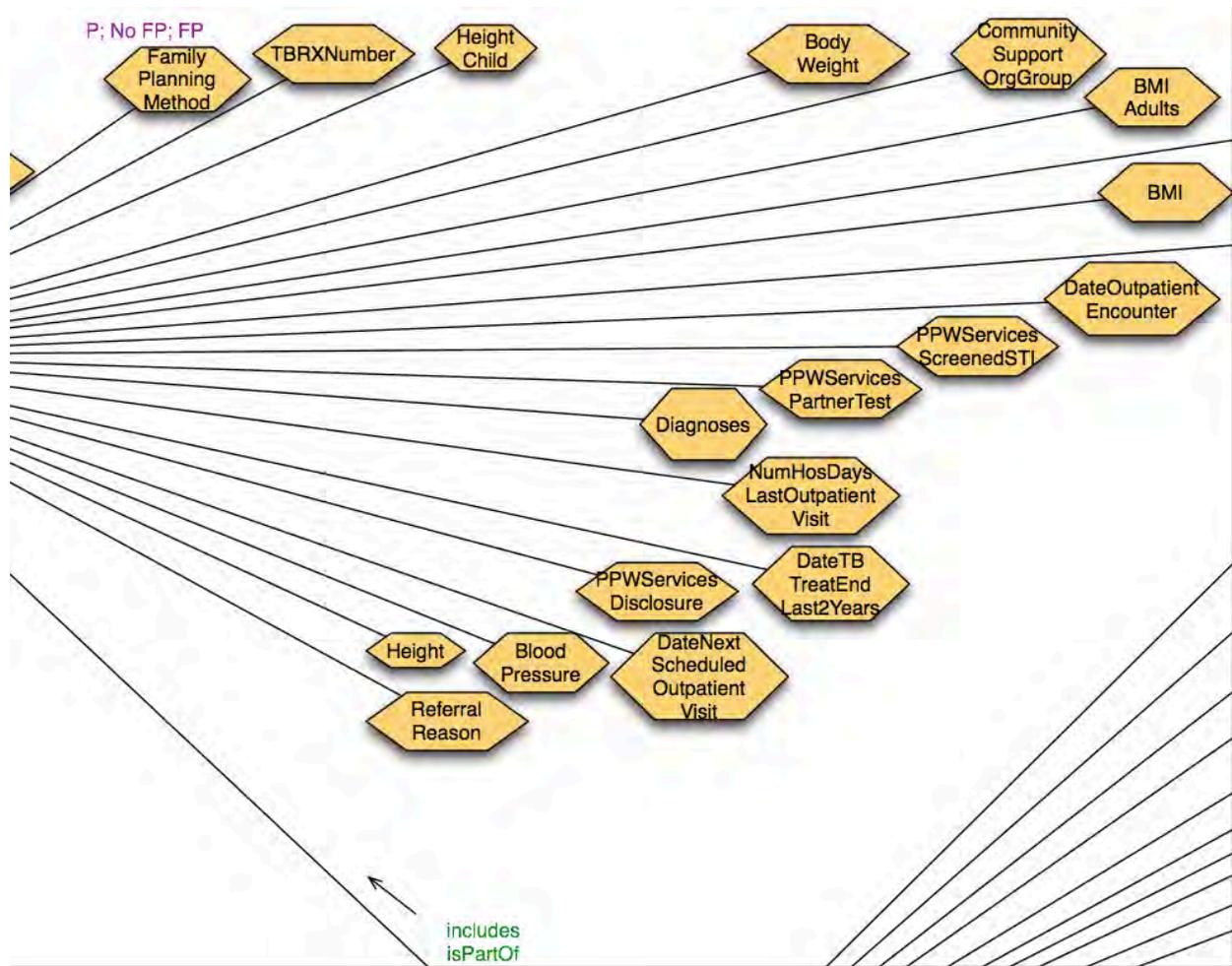
148	Nearest Healthcare Facility		x					
149	Nearest Landmark		x					
150	Nutritional Support Needed							x
151	Diflucan							x
152	Basic HIV education, transmission	x	x		x		x	x
153	Prevention: abstinence, safer sex, condoms	x	x		x		x	x
154	Prevention: household precautions, what is safe	x	x		x		x	x
155	Post-test counseling: implications of results	x	x		x		x	x
156	Positive living	x	x		x		x	x
157	Testing partners-Discordants	x	x		x		x	x
158	Disclosure	x	x		x		x	x
159	To whom disclosed (list)	x	x		x		x	x
160	Family/living situation	x	x		x		x	x
161	Shared confidentiality	x	x		x		x	x
162	Reproductive choices, prevention MTCT	x	x		x		x	x
163	Child's blood test	x	x		x		x	x
164	Progression of disease	x	x		x		x	x
165	Available treatment/prophylaxis	x	x		x		x	x
166	Follow-up apointments, clinical team	x	x		x		x	x
167	CTX, INH prophylaxis	x	x		x		x	x
168	CTX, prophylaxis				x		x	
169	Malaria prevention, IPT, ITN		x					
170	ART -- educate on essentials (locally adapted)	x	x		x		x	x
171	Why complete adherence needed	x	x		x		x	x
172	Adherence preparation, indicate visits	x	x		x		x	x
173	Indicate when READY for ART: Date/result Clinical team discussion	x	x		x		x	x
174	Explain dose, when to take	x	x		x		x	x
175	What can occur, how to manage side effects	x	x		x		x	x
176	What to do if one forgets dose	x	x		x		x	x
177	What to do when travelling	x	x		x		x	x
178	Nutritional information						x	
179	Adherence plan (schedule, aids, explain diary)	x	x		x		x	x
180	Treatment supporter preparation	x	x		x		x	x
181	Which does, why missed	x	x		x		x	x
182	ARV support group	x	x		x		x	x
183	How to contact clinic	x	x		x		x	x
184	Symptom management, palliative care at home	x	x		x		x	x
185	Caregiver booklet	x	x		x		x	x
186	Home-based care --specify	x	x		x		x	x
187	Support groups	x	x		x		x	x
188	Community support	x	x		x		x	x
189	At 6, 12, 24 months Date					x		
190	At 6, 12, 24 months WHO Clinical Stage					x		
191	At 6, 12, 24 months Weight					x		

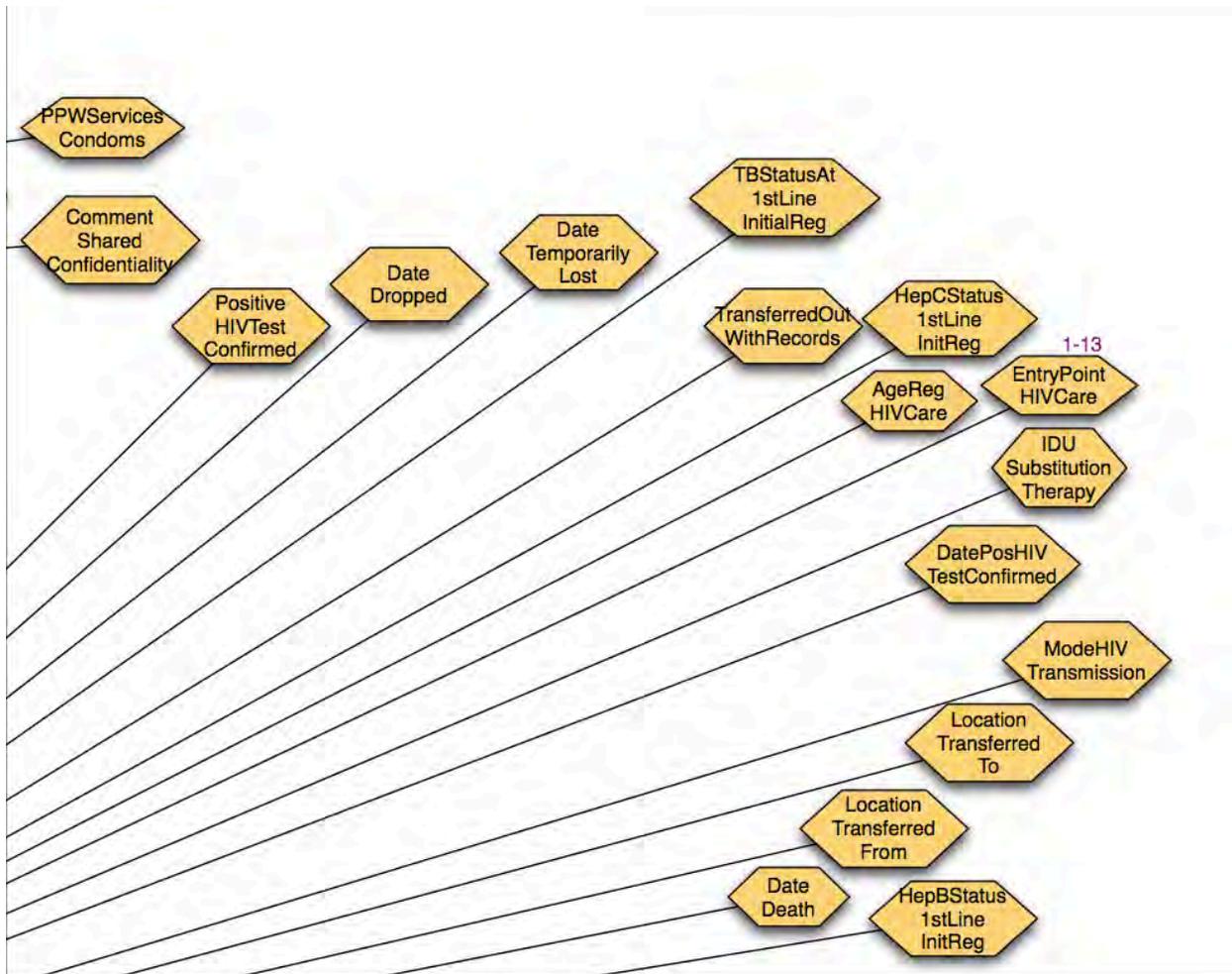
192	At 6, 12, 24 months Height					x		
193	At 6, 12, 24 months TLC					x		
194	At 6, 12, 24 months CD4					x		
195	Mode of HIV transmission					x		
196	Literate					x		
197	Employed					x		
198	Alcoholism					x		
199	Estimated monthly household income					x		
200	Community support organization/group							x

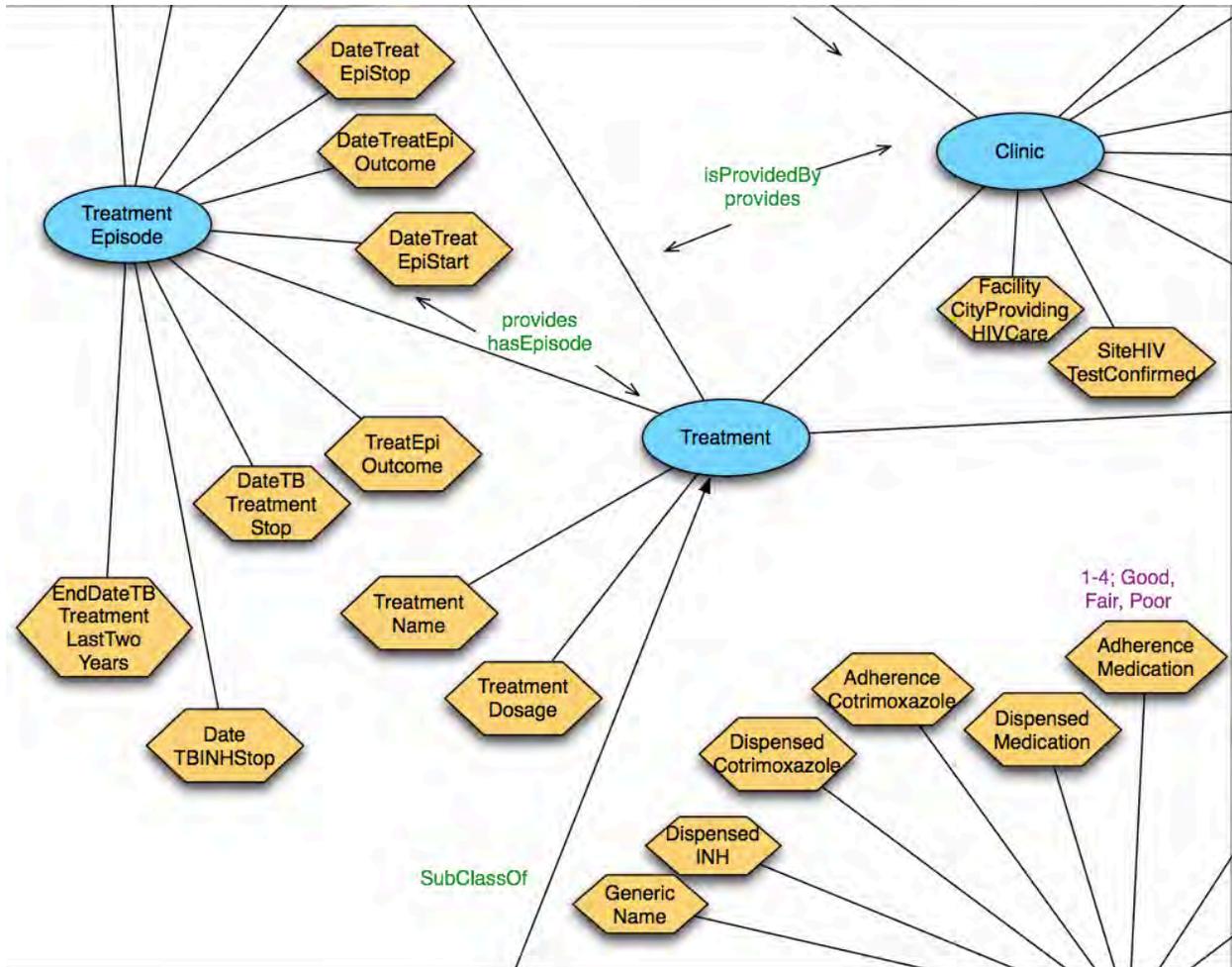


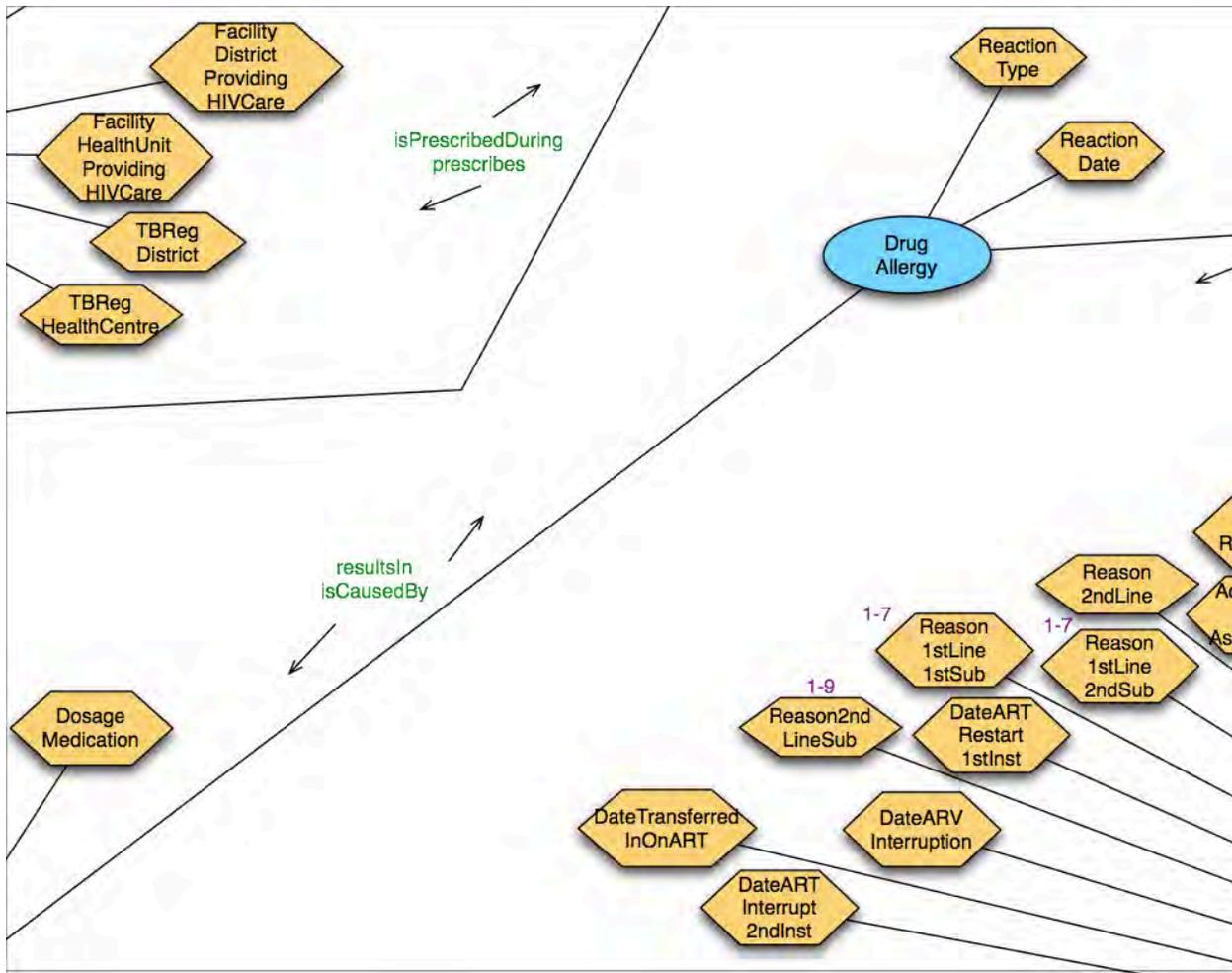


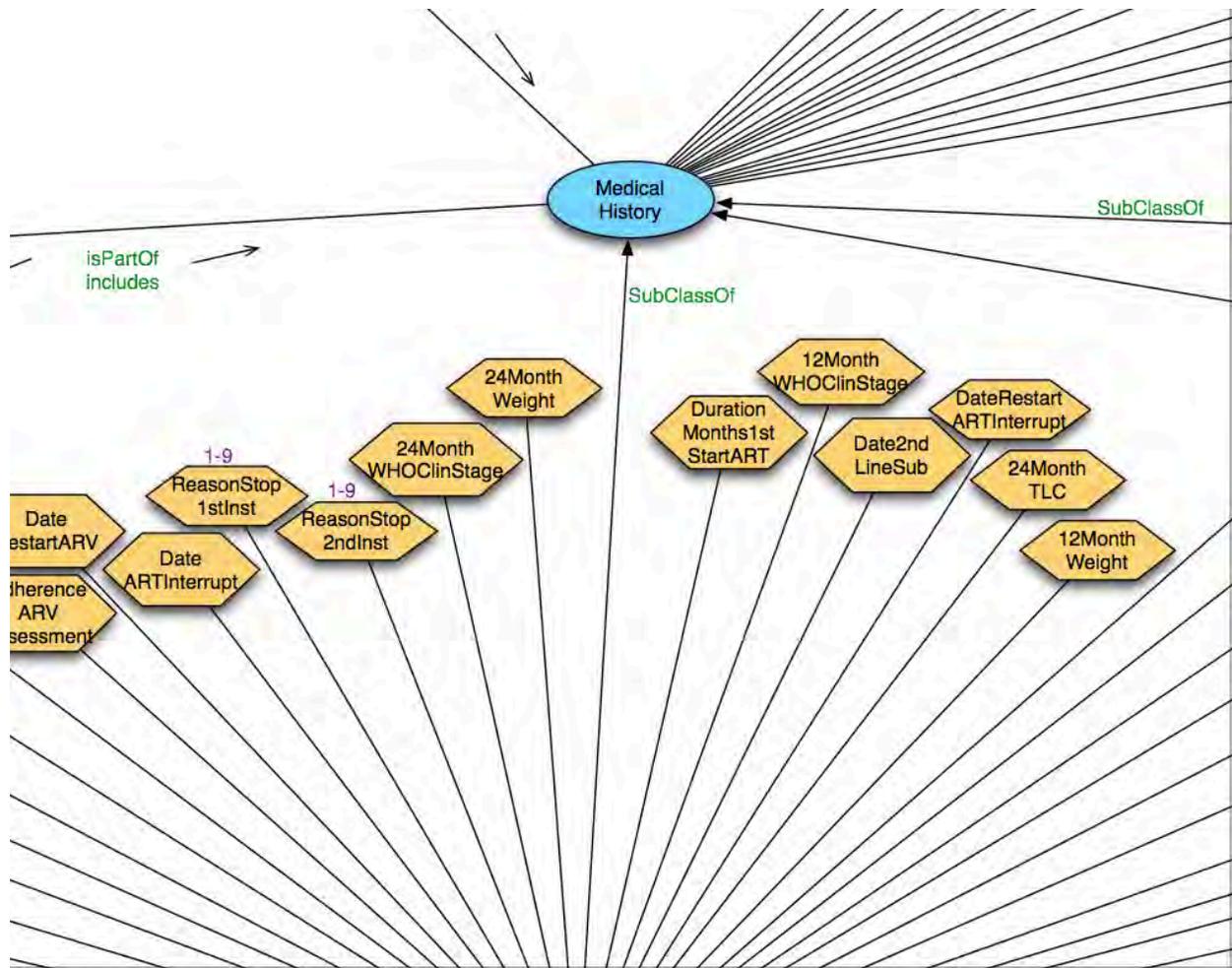


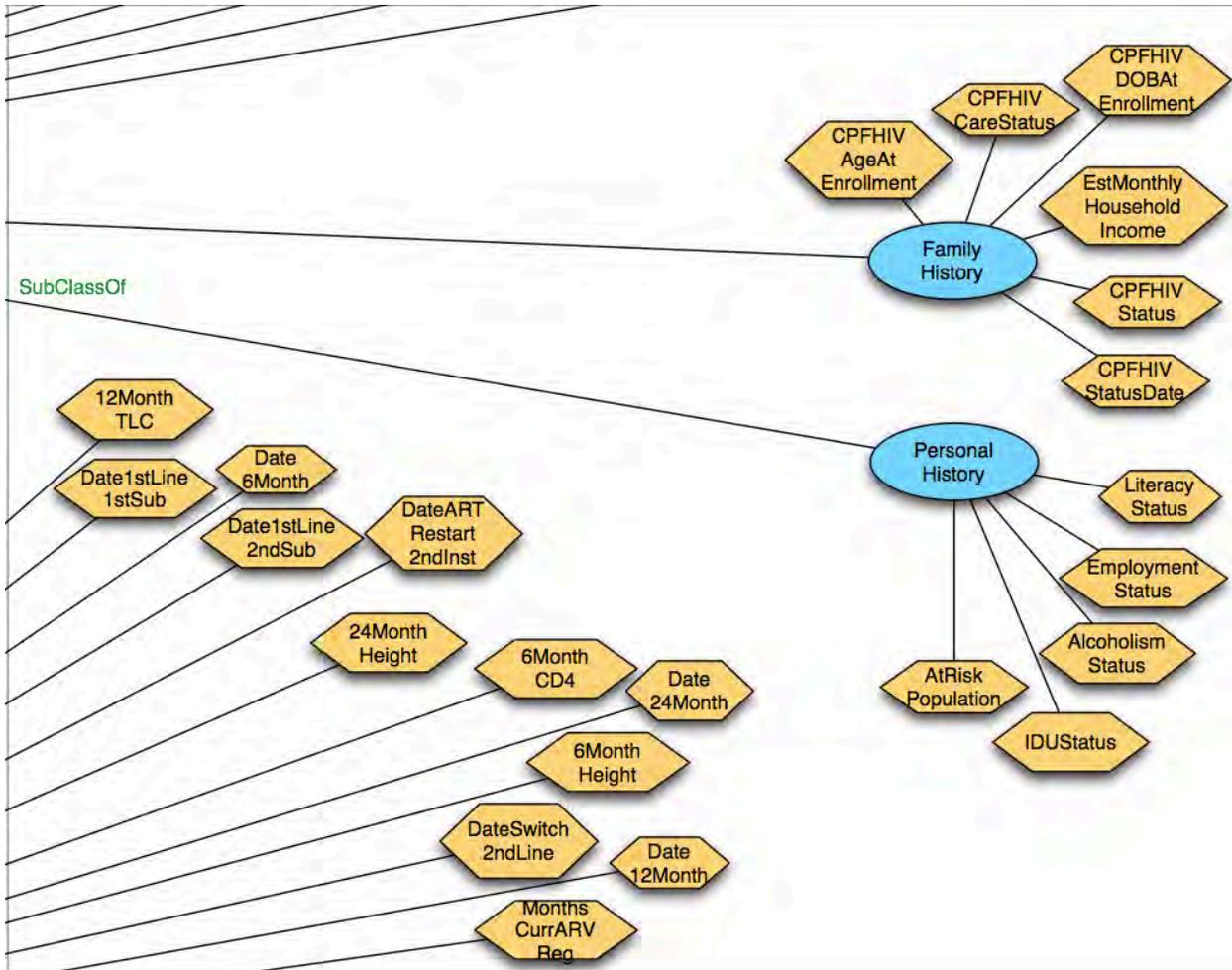


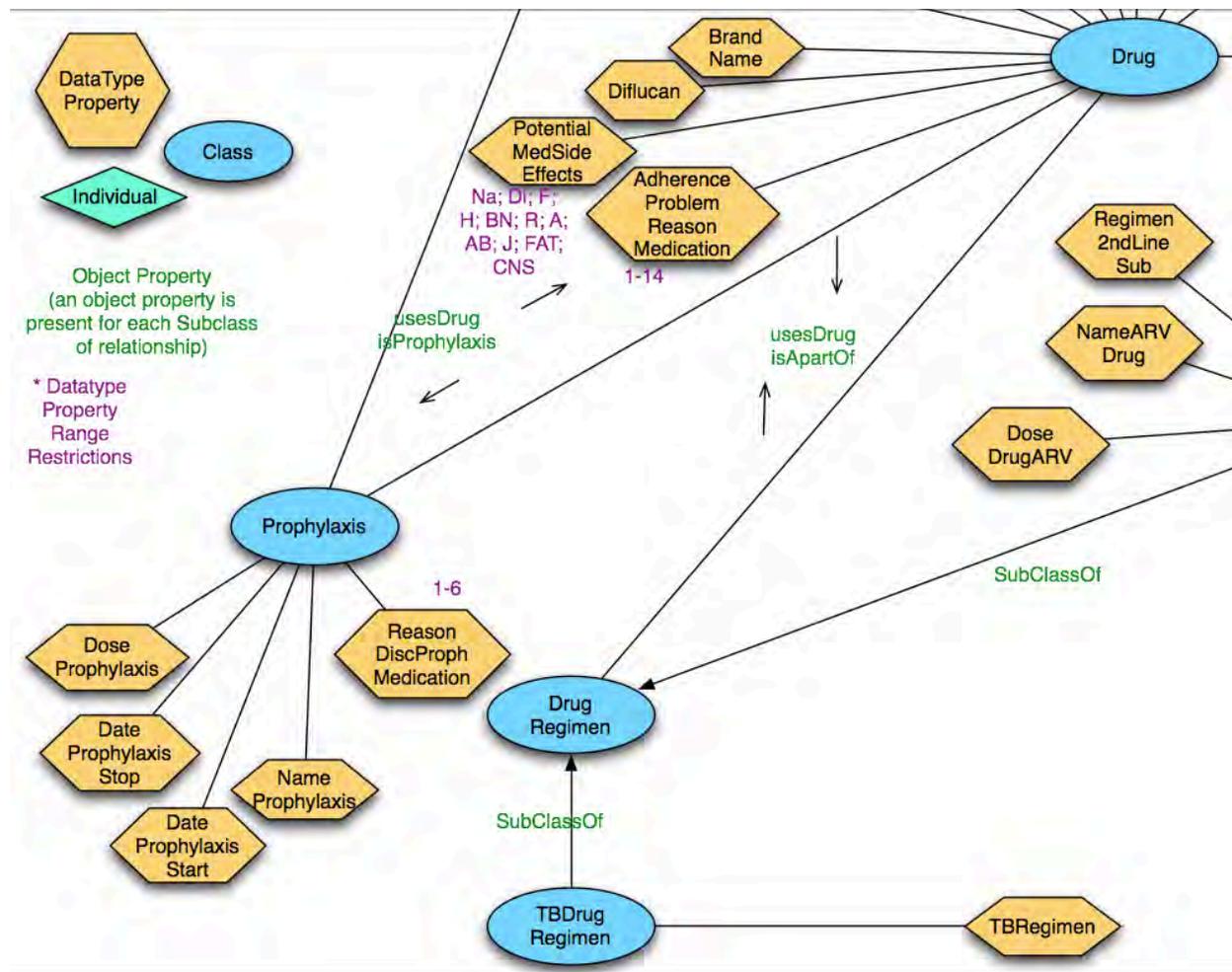


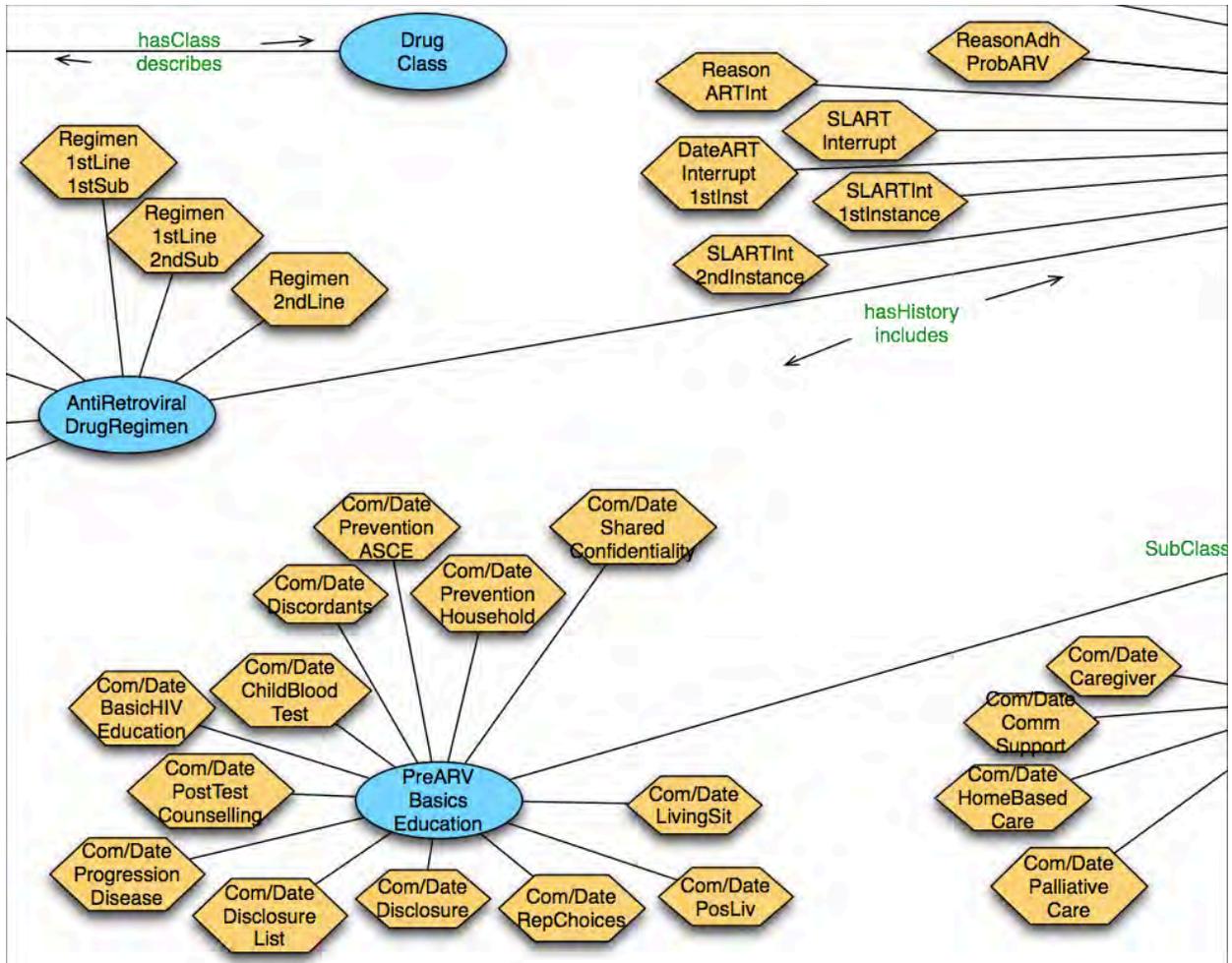


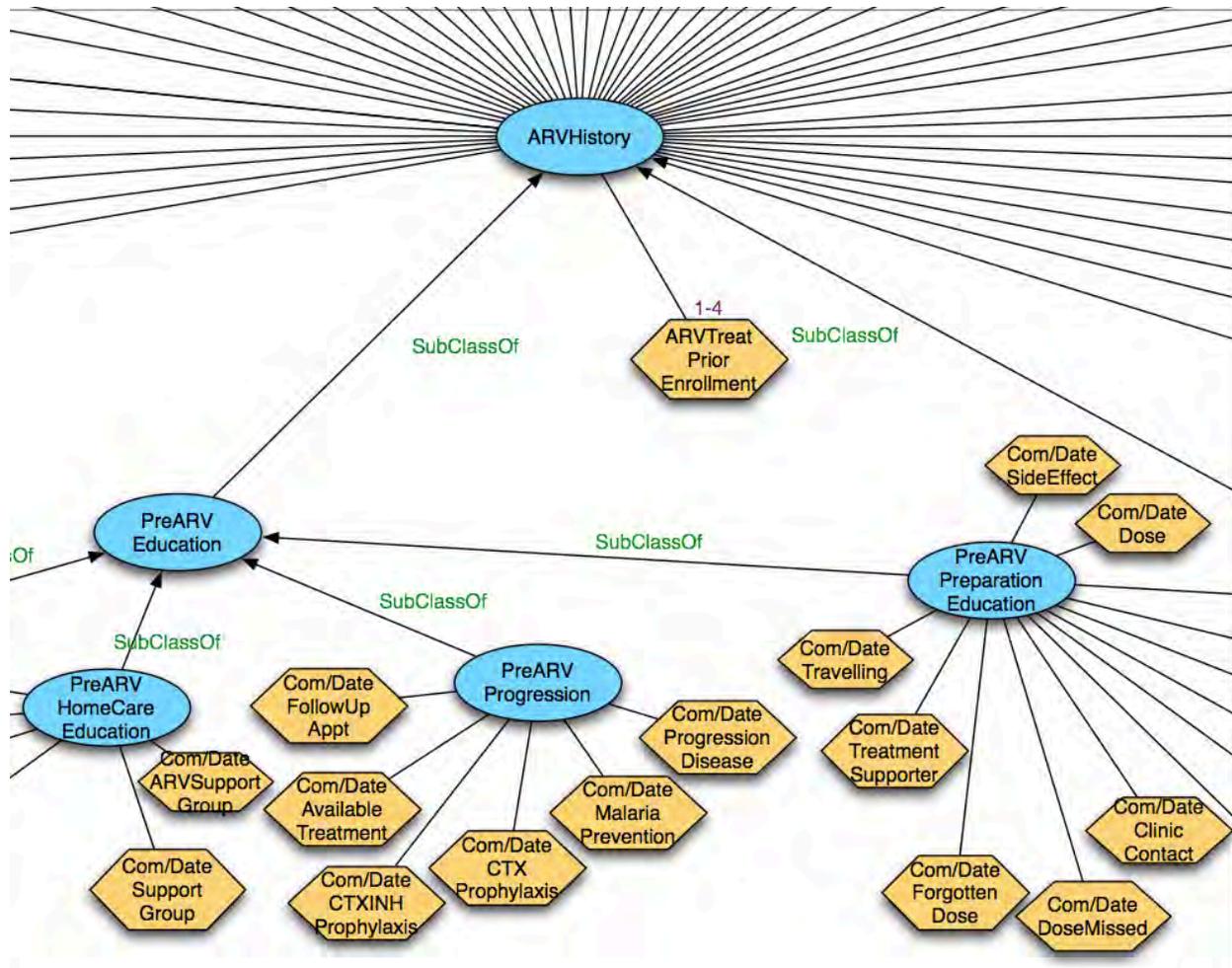


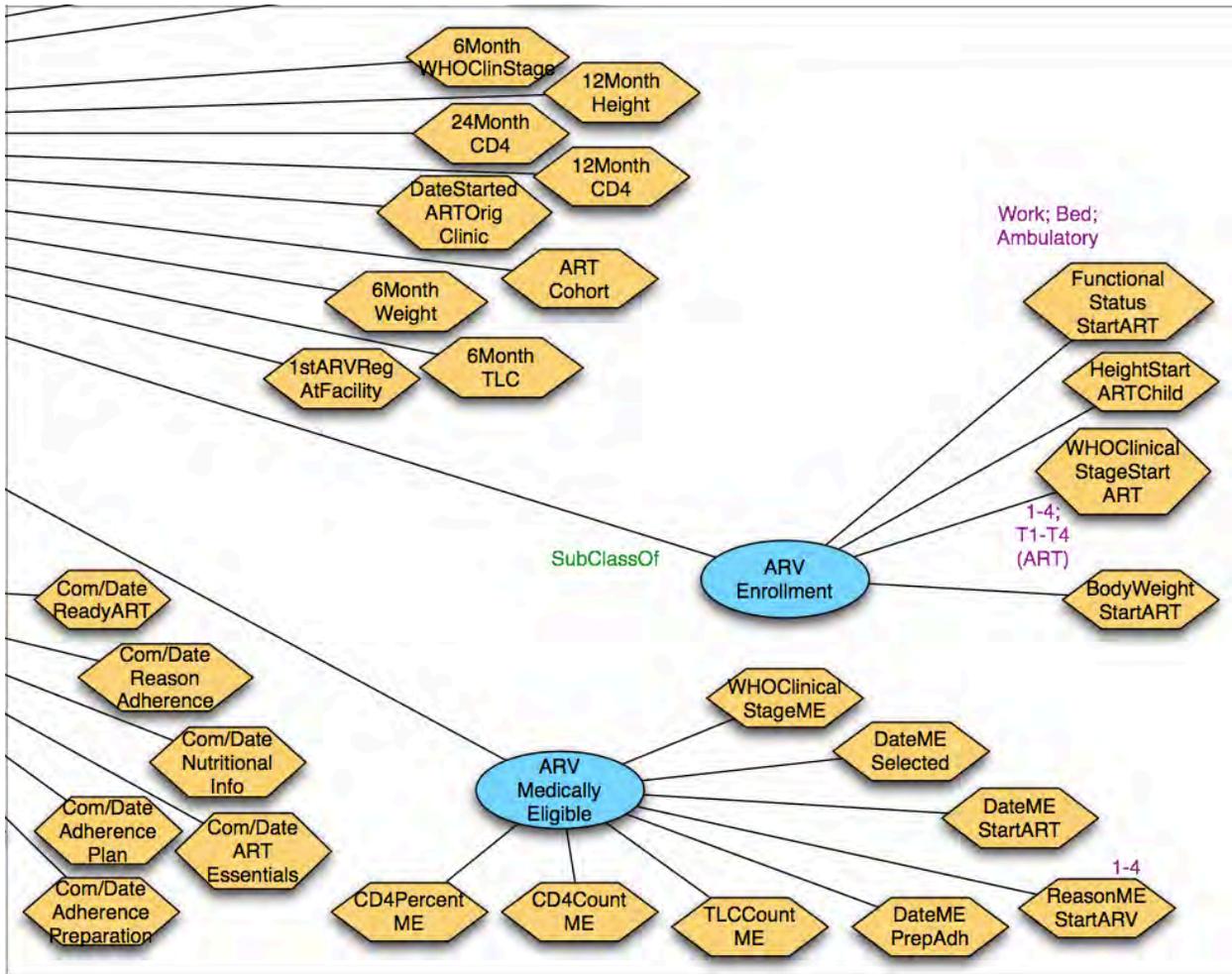












## Appendix H. System screenshots by scenario

**Scenario 1: Querying other provider's systems to identify prior medical records for a given patient.**

**Query 1: Find all encounters for Patient X**

Screenshots for this query are included in Chapter 4.

**Query 2: Find all future visit dates for Patient X**

[Pt ID Query](#) [Cohort Query](#) [Scenario-Based Queries](#) [About](#) [Logout](#)

### Scenario 1, Query 2: Find all future visit dates for Patient X.

Patient Attributes	
Clinic ID:	56364
First Name:	Jasper
Middle Name:	
Last Name:	Ngetch
Age*:	1-100
Date of Birth:	05-06-1926 <input type="text"/> (dd-mm-yyyy) <small>If month and/or day unknown please enter "01".</small>
Gender:	<input checked="" type="radio"/> Male <input type="radio"/> Female <input type="radio"/> Male & Female
Encounter Attributes	
Visit Date:	date range on/before <input type="text"/> (dd-mm-yyyy) on/after 01-01-2012 <input type="text"/> (dd-mm-yyyy)
<input type="button" value="Reset"/> <input type="button" value="Submit Query"/>	

\*Age is calculated based on current date.

Figure 31 - Scenario 1, Query 2 Find all future visit dates for Patient X. Query entry form Screenshot.

[Pt ID Query](#) [Cohort Query](#) [Scenario-Based Queries](#) [About](#) [Logout](#)

### Cohort Query Results

Query Parameters: Gender: ; [Edit Query](#)

ClinicID	Name Prefix	First Name	Middle Name	Last Name Prefix	Last Name	Last Name 2	Last Name Suffix	Gender	Birthdate	Age*	Status	Clinic ID Matches	Probability	Show Comparison
<a href="#">56364</a>		Jasper			Ngetch			M	05-06-1926	86				

\*Age is calculated based on current date.

Figure 32 - Scenario 1, Query 2 Patient Query Results Screenshot

Query 3: Identify recent laboratory data for Patient X

[Pt ID Query](#) [Cohort Query](#) [Scenario-Based Queries](#) [About](#) [Logout](#)

### Scenario 1, Query 3: Identify recent laboratory data for Patient X.

Patient Attributes	
Clinic ID:	56364
First Name:	Jasper
Middle Name:	
Last Name:	Ngetch
Age*:	1-100
Date of Birth:	05-06-1926 (dd-mm-yyyy) If month and/or day unknown please enter "01".
Gender:	<input checked="" type="radio"/> Male <input type="radio"/> Female <input type="radio"/> Male & Female
Patient Status:	<input checked="" type="radio"/> Alive Only <input type="radio"/> Deceased Only <input type="radio"/> Alive or Deceased
Encounter Attributes	
Observations:	Weight not
	date range on/before 01-01-1990 (dd-mm-yyyy)
	on/after 31-12-2012 (dd-mm-yyyy)
	CD4 Count not
	date range on/before 01-01-1990 (dd-mm-yyyy)
	on/after 31-12-2012 (dd-mm-yyyy)
	not
	date range on/before (dd-mm-yyyy)
	on/after (dd-mm-yyyy)
	not
	date range on/before (dd-mm-yyyy)
	on/after (dd-mm-yyyy)
<input type="button" value="Reset"/> <input type="button" value="Submit Query"/>	

\*Age is calculated based on current date.

Figure 33 - Scenario 1, Query 3: Identify recent laboratory data for Patient X, Query entry form screenshot

### Cohort Query Results

Query Parameters: Gender :

[Edit Query](#)

ClinicID	Name Prefix	First Name	Middle Name	Last Name Prefix	Last Name	Last Name 2	Last Name Suffix	Gender	Birthdate	Age*	Status	Clinic ID Matches	Probability	Show Comparison
<a href="#">161024</a>		Jasper			Ngetch			M	1926-06-05	86				
<a href="#">161024</a>		Jasper			Ngetch			M	1926-06-05	86				
<a href="#">161024</a>		Jasper			Ngetch			M	1926-06-05	86				
<a href="#">161026</a>		Jasper			Ngetch			M	1926-06-05	86				
<a href="#">161026</a>		Jasper			Ngetch			M	1926-06-05	86				
<a href="#">161035</a>		Jasper			Ngetch			M	1926-06-05	86				
<a href="#">161035</a>		Jasper			Ngetch			M	1926-06-05	86				

\*Age is calculated based on current date.

Figure 34 - Scenario 1, Query 3 Patient Query Results Screenshot

**Scenario 2: Determining the inclusion criteria for clinical trials, and querying systems to determine patients who will be included or excluded from the trial**

[Pt ID Query](#) [Cohort Query](#) [Scenario-Based Queries](#) [About](#) [Logout](#)

### Cohort Query

Scenario 2: Determining the inclusion criteria for clinical trials, and querying systems to determine patients who will be included or excluded from the trial.)

---

**Patient Attributes**

Gender:  Male  Female  Male & Female

Age\*: between 20 and 30 years

Birthdate: between and

Patient Status:  Alive Only  Deceased Only  Alive or Deceased

**Encounter Attributes**

Diagnosis: equal None date range on/before on/after

Drugs: equal None date range on/before on/after

Observations: None not date range on/before on/after

WHO Stage: equal Adult Stage 2

Visit Date: date range on/before 01-01-1996 on/after 31-12-2012

Physician's Name: equal

Clinic Name: equal

\*Age is calculated based on current date.

**Figure 35 - Scenario 2 Cohort Query Entry Form Screenshot**

[Pt ID Query](#) [Cohort Query](#) [Scenario-Based Queries](#) [About](#) [Logout](#)

### Cohort Query Results

Query Parameters: Gender: ; [Edit Query](#)

ClinicID	Name Prefix	First Name	Middle Name	Last Name Prefix	Last Name	Last Name 2	Last Name Suffix	Gender	Birthdate	Age*	Status	Clinic ID Matches	Probability	Show Comparison
<a href="#">61979</a>		Maya			Kibenei			F	2062-05-10	-50				
<a href="#">51742</a>		Charlie			Sang			M	1974-11-19	38				
<a href="#">42940</a>		Callum			Chemurtoi			M	2019-05-13	-7				
<a href="#">54460</a>		Gracie			Mugun			F	2065-04-19	-53				

\*Age is calculated based on current date.

**Figure 36 - Scenario 2 Cohort Query Results Screenshot**

Scenario 3: Identifying adverse events, routine monitoring, side effects or contraindications caused by routine clinical care or clinical trial study protocols.

Query 1: Find all patients of Physician Y who are on ART and have not had a clinical encounter in the past 3 months.

[Pt ID Query](#) [Cohort Query](#) [Scenario-Based Queries](#) [About](#) [Logout](#)

**Scenario 3, Query 1 & Query 2: Find all patients of Physician/Clinic Y who are on drug(s) A and have/have not had a clinical encounter in the past X months.**

Patient Attributes	
Patient Status:	<input type="radio"/> Alive Only <input type="radio"/> Deceased Only <input checked="" type="radio"/> Alive or Deceased
Encounter Attributes	
Drugs:	equal ▾ 1a(30)-STAVUIDINE (3C date range on/before <input type="text"/> (dd-mm-yyyy) on/after <input type="text"/> (dd-mm-yyyy)
	equal ▾ <input type="text"/> date range on/before <input type="text"/> (dd-mm-yyyy) on/after <input type="text"/> (dd-mm-yyyy)
	equal ▾ <input type="text"/> date range on/before <input type="text"/> (dd-mm-yyyy) on/after <input type="text"/> (dd-mm-yyyy)
	equal ▾ <input type="text"/> date range on/before <input type="text"/> (dd-mm-yyyy) on/after <input type="text"/> (dd-mm-yyyy)
Visit Date:	date range on/before <input type="text"/> (dd-mm-yyyy) on/after 01-07-2012 <input type="text"/> (dd-mm-yyyy)
Physician's Name:	equal ▾ Brian Wilson <input type="text"/>
Clinic Name:	equal ▾ <input type="text"/>
<input type="button" value="Reset"/> <input type="button" value="Submit Query"/>	

\*Age is calculated based on current date.

Figure 37 - Scenario 3, Query 1 Find all patients of Physician Y who are on drug(s) A and have/have not had a clinical encounter in the past X months. Query entry form screenshot.

### Cohort Query Results

Query Parameters: Gender ;

[Edit Query](#)

ClinicID	Name Prefix	First Name	Middle Name	Last Name Prefix	Last Name	Last Name 2	Last Name Suffix	Gender	Birthdate	Age*	Status	Clinic ID Matches	Probability	Show Comparison
<a href="#">53091</a>		Riley			Shiv			M	2052-02-26	-40				
<a href="#">64710</a>		Emily			Waula			F	1981-01-21	32				
<a href="#">60052</a>		Reuben			Simani			M	2058-06-28	-46				
<a href="#">45483</a>		Sam			Lokonye			M	1993-01-15	20				
<a href="#">51569</a>		Jasper			Chubla			M	2016-12-21	-4				
<a href="#">59068</a>		Ben			Isiaho			M	2014-12-27	-2				
<a href="#">49193</a>		Jackson			Chepleting			M	2020-10-17	-8				
<a href="#">67695</a>		Isobel			Chepkemboi			F	1980-02-09	32				
<a href="#">45456</a>		Nathaniel			Surtoi			M	2042-08-07	-30				
<a href="#">59512</a>		Zara			Letio			F	2021-06-22	-9				
<a href="#">45549</a>		Grace			Legai			F	2048-03-17	-36				
<a href="#">41503</a>		Molly			Mogire			F	2060-08-02	-48				
<a href="#">63718</a>		Harvey			Kosabei			M	2020-12-08	-8				
<a href="#">57431</a>		Lucas			Machanga			M	1980-12-02	32				
<a href="#">50789</a>		Harrison			Kipkemei			M	2016-10-10	-4				
<a href="#">51877</a>		Laila			Kiptoom			F	1970-05-31	42				
<a href="#">57567</a>		Edward			Kipchirchir			M	1988-11-21	24				
<a href="#">69716</a>		Amy			Ngesirei			F	2059-02-06	-47				

Figure 38 - Scenario 3, Query 1 Physician Cohort Results Screenshot

Query 2: Find all patients of Clinic Z, or drug regimen ABC and DEF during time period G.

[Pt ID Query](#) [Cohort Query](#) [Scenario-Based Queries](#) [About Logout](#)

**Scenario 3, Query 1 & Query 2: Find all patients of Physician/Clinic Y who are on drug(s) A and have/have not had a clinical encounter in the past X months.**

**Patient Attributes**

**Patient Status:**  Alive Only  Deceased Only  Alive or Deceased

**Encounter Attributes**

**Drugs:**

equal : ZIDOVDINE/LAMIVUDI date range on/before (dd-mm-yyyy) on/after (dd-mm-yyyy)

equal : D4T(30)/3TC/NVP date range on/before (dd-mm-yyyy) on/after (dd-mm-yyyy)

equal : D4T(40)/3TC/EFV date range on/before (dd-mm-yyyy) on/after (dd-mm-yyyy)

equal : date range on/before (dd-mm-yyyy) on/after (dd-mm-yyyy)

**Visit Date:** date range on/before 01-01-2000 (dd-mm-yyyy) on/after 01-12-2012 (dd-mm-yyyy)

**Physician's Name:** equal :

**Clinic Name:** equal : AAR Nakuru Clinic

\*Age is calculated based on current date.

Figure 39 - Scenario 3, Query 2 Clinic Query Results Screenshot

[Pt ID Query](#) [Cohort Query](#) [Scenario-Based Queries](#) [About Logout](#)

**Cohort Query Results**

Query Parameters: Gender: ; [Edit Query](#)

ClinicID	Name Prefix	First Name	Middle Name	Last Name Prefix	Last Name	Last Name 2	Last Name Suffix	Gender	Birthdate	Age*	Status	Clinic ID Matches	Probability	Show Comparison
<a href="#">61979</a>		Maya			Kibenei			F	2062-05-10	-50				
<a href="#">51742</a>		Charlie			Sang			M	1974-11-19	38				
<a href="#">42940</a>		Callum			Chemurtoi			M	2019-05-13	-7				
<a href="#">54460</a>		Gracie			Mugun			F	2065-04-19	-53				

\*Age is calculated based on current date.

Figure 40 - Scenario 3, Query 2 Clinic Query Results Screenshot

## Appendix I. Installation Instructions

1. System Configuration
  - Modify sysConfig.properties file
2. Modify mapping documents
  - uiMappingDoc.xml
  - dbOntMapDocPI\_SystemName.xml
  - dbOntMapDocPtID\_SystemName.xml
  - dbOntMapDocHIV\_SystemName.xml
  - dbOntMapDocHIV\_SystemName.xml
3. Install Firefox
4. Install Java
5. Install Tomcat (default settings)
6. Start Tomcat
7. Download OBDIS.war
8. Navigate to <http://localhost:8080/manager/html> and enter your Tomcat administrator username and password
9. In the Tomcat Web Application Manager, enter the location of the downloaded openmrs.war file to deploy
10. The Tomcat page will refresh and /obdis should be displayed under Applications. Apache Tomcat should also start the application automatically.
11. Add configuration file (#1) and mapping documents (#2) to the Resources directory of the application (/obdis/customJava/resources).
12. Navigate to <http://localhost:8080/obdis> or refresh the OBDIS page Tomcat opened
13. Login using the following username and password:
  - username: admin
  - password: admin123

## Appendix J. Manual and Generated SQL Queries by Scenario

### Simulated Evaluation

**Scenario 1: Querying other provider's systems to identify prior medical records for a given patient.**

**Query 1: Find all encounters for Patient X**

**Final OpenMRS Simulated:**

```
SELECT enc.encounter_id,trim(date(enc.encounter_datetime)), et.name as EncType, loc.name as Clinic,
p.gender, p.birthdate, FLOOR(((DATEDIFF(CURDATE(), birthdate))/365)) as Age, pn.prefix,
pn.given_name, pn.middle_name,pn.family_name_prefix, pn.family_name, pn.family_name2,
pn.family_name_suffix, enc.patient_id FROM encounter enc JOIN encounter_type et
ON enc.encounter_type = et.encounter_type_id LEFT OUTER JOIN location loc ON loc.name =
enc.location_id JOIN person p ON p.person_id = enc.patient_id JOIN person_name pn ON pn.person_id
= enc.patient_id JOIN patient_identifier pi ON pi.patient_id = pn.person_id WHERE pi.identifier = '56364'
ORDER BY enc.encounter_datetime desc
```

**Final OpenMRS Manual:**

```
SELECT pi.identifier, pn.given_name, pn.middle_name, pn.family_name_prefix, pn.family_name,
pn.family_name2, pn.family_name_suffix, pn.family_name_suffix, pn.degree, p.gender, p.birthdate,
FLOOR(((DATEDIFF(CURDATE(), p.birthdate))/365)) as Age, enc.encounter_id, et.name FROM person
p, person_name pn, patient_identifier pi, encounter enc, encounter_type et WHERE pn.person_id =
p.person_id AND p.gender = 'M' AND p.birthdate = '1926-06-05' AND pn.given_name = 'Jasper' AND
pn.family_name = 'Ngetch' AND pi.identifier = '56364' AND pi.patient_id = p.person_id AND
enc.patient_id = p.person_id AND enc.encounter_type = et.encounter_type_id
```

**Final OpenClinica Simulated:**

```
SELECT s.subject_id, s.unique_identifier, ss.study_subject_id, se.study_event_id, se.location,
date(se.date_start), sed.name, sed.type FROM subject s, study_subject ss, study_event se,
study_event_definition sed WHERE ss.study_subject_id = '2'AND s.subject_id = ss.study_subject_id
AND ss.study_subject_id = se.study_subject_id AND se.study_event_definition_id =
sed.study_event_definition_id ORDER BY date
```

**Final OpenClinica Manual:**

```
SELECT s.subject_id, s.unique_identifier, ss.study_subject_id, se.study_event_id, se.location,
date(se.date_start), sed.name, sed.type FROM subject s, study_subject ss, study_event se,
study_event_definition see WHERE ss.study_subject_id = '2'AND s.unique_identifier = '56008' and
s.subject_id = ss.study_subject_id AND ss.study_subject_id = se.study_subject_id AND
se.study_event_definition_id = sed.study_event_definition_id ORDER BY date
```

**Query 2: Find all future visit dates for Patient X**

**Final OpenMRS Query Simulated:**

```
SELECT enc.encounter_id, trim(date(enc.encounter_datetime)), et.name as EncType, loc.name as Clinic,
p.gender, p.birthdate, FLOOR(((DATEDIFF(CURDATE(), birthdate))/365)) as Age, pn.prefix,
pn.given_name, pn.middle_name,pn.family_name_prefix, pn.family_name, pn.family_name2,
pn.family_name_suffix, enc.patient_id FROM encounter enc JOIN encounter_type et
ON enc.encounter_type = et.encounter_type_id LEFT OUTER JOIN location loc ON loc.name =
enc.location_id JOIN person p ON p.person_id = enc.patient_id JOIN person_name pn ON pn.person_id
= enc.patient_id JOIN patient_identifier pi ON pi.patient_id = pn.person_id WHERE pi.identifier = '56364'
ORDER BY enc.encounter_datetime desc
```

**Final OpenMRS Query Manual:**

```
SELECT pi.identifier, pn.given_name, pn.middle_name, pn.family_name_prefix, pn.family_name,
pn.family_name2, pn.family_name_suffix, pn.family_name_suffix, pn.degree, p.gender,
p.birthdate, FLOOR(((DATEDIFF(CURDATE(), p.birthdate))/365)) as Age, enc.encounter_id, et.name,
obs.value_datetime FROM person p, person_name pn, patient_identifier pi, encounter enc,
encounter_type et, obs WHERE pn.person_id = p.person_id AND p.gender = 'M' AND p.birthdate =
'1926-06-05' AND pn.given_name = 'Jasper' AND pn.family_name = 'Ngetch' AND pi.identifier = '56364'
AND pi.patient_id = p.person_id AND enc.patient_id = p.person_id AND enc.encounter_type =
et.encounter_type_id AND obs.person_id = p.person_id AND obs.concept_id = 5096 and
obs.value_datetime > '2012-01-01'
```

**Query 3: Identify recent laboratory data for Patient X****Final OpenMRS Query Simulated:**

```
SELECT distinct(pi.identifier) as ptid, pn.prefix, pn.given_name, pn.middle_name, pn.family_name_prefix,
pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender, DATE_FORMAT(p.birthdate, '%d-
%m-%Y') as birthdate, FLOOR(((DATEDIFF(CURDATE(), p.birthdate))/365)) as Age, p.dead,
question.concept_id, question.name, obs.value_numeric, answer.name FROM concept_name question,
obs, concept c, concept_name answer, person p, person_name pn, patient_identifier pi WHERE
obs.concept_id = question.concept_id and answer.concept_id = obs.concept_id and obs.concept_id =
c.concept_id AND question.locale = 'en' AND answer.locale = 'en' and p.person_id = pn.person_id and
pi.patient_id = p.person_id and obs.person_id = p.person_id and c.class_id in (1,2) AND p.dead = 0 AND
pn.given_name = 'Jasper' AND pn.family_name = 'Ngetch' AND p.birthdate = '1926-06-05' AND
pi.identifier = '56364' AND question.concept_id in (161024, 161026, 161035) AND obs.obs_datetime
BETWEEN '1990-01-01' AND '2012-12-31' AND question.concept_id in (161024, 161026, 161035) AND
obs.obs_datetime BETWEEN '1990-01-01' AND '2012-12-31' AND question.concept_id in (161024,
161026, 161035) AND obs.obs_datetime BETWEEN '1990-01-01' AND '2012-12-31' AND
question.concept_id in (161024, 161026, 161035) AND obs.obs_datetime BETWEEN '1990-01-01' AND
'2012-12-31' GROUP BY question.concept_id, question.name, obs.value_numeric ORDER BY
obs.obs_datetime
```

**Final OpenMRS Query Manual:**

```
SELECT distinct(p.person_id) as ptid, pn.prefix, pn.given_name, pn.middle_name,
pn.family_name_prefix, pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender,
DATE_FORMAT(p.birthdate, '%d-%m-%Y') as birthdate, FLOOR(((DATEDIFF(CURDATE()),
p.birthdate))/365)) as Age, p.dead, question.concept_id, question.name, obs.value_numeric,
answer.name FROM concept_name question, obs, concept c, concept_name answer, person
p, person_name pn, patient_identifier pi WHERE obs.concept_id = question.concept_id and
answer.concept_id = obs.concept_id and obs.concept_id = c.concept_id AND question.locale = 'en' AND
answer.locale = 'en' and p.person_id = pn.person_id and pi.patient_id = p.person_id and obs.person_id =
p.person_id and c.class_id in (1,2) AND p.dead = 0 AND p.gender = 'M' AND pn.given_name = 'Jasper'
AND pn.family_name = 'Ngetch' AND p.birthdate = '1926-06-05' AND pi.identifier = '56364' AND
question.concept_id in (161024, 161026, 161035) AND obs.obs_datetime BETWEEN '1990-01-01' AND
'2013-01-01' GROUP BY question.concept_id, question.name, obs.value_numeric ORDER BY
obs.obs_datetime
```

**Scenario 2: Determining the inclusion criteria for clinical trials, and querying systems to determine patients who will be included or excluded from the trial****Final OpenMRS Simulated:**

```
SELECT p.person_id as ptid, pn.prefix, pn.given_name, pn.middle_name, pn.family_name_prefix,
pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender, p.birthdate,
FLOOR(((DATEDIFF(CURDATE(), p.birthdate))/365)) as Age, p.dead, pi.identifier FROM person p,
person_name pn, concept_name question, encounter enc, patient_identifier pi, obs LEFT OUTER JOIN
concept_name answer ON (answer.concept_id = obs.value_coded) WHERE obs.concept_id =
question.concept_id AND obs.person_id = p.person_id AND p.person_id = pn.person_id AND obs.voided
!= 1 AND answer.locale = 'en' AND question.locale = 'en' AND enc.encounter_id = obs.encounter_id AND
```

```
pi.patient_id = p.person_id AND p.dead = 0 AND p.birthdate BETWEEN '1982-01-01' AND '1992-01-01'
AND enc.encounter_datetime BETWEEN '1996-01-01' AND '2012-12-31' AND answer.concept_id = 1205
GROUP BY obs.person_id
```

#### Final OpenMRS Manual:

```
SELECT pi.identifier, pn.prefix, pn.given_name, pn.middle_name, pn.family_name_prefix,
pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender, p.birthdate,
FLOOR(((DATEDIFF(CURDATE(), p.birthdate))/365)) as Age, p.dead, pi.identifier FROM person p,
person_name pn, concept_name question, encounter enc, patient_identifier pi, obs LEFT OUTER JOIN
concept_name answer ON (answer.concept_id = obs.value_coded) WHERE obs.concept_id =
question.concept_id AND obs.person_id = p.person_id AND p.person_id = pn.person_id AND obs.voided
!= 1 AND answer.locale = 'en' AND question.locale = 'en' AND enc.encounter_id = obs.encounter_id AND
pi.patient_id = p.person_id AND p.dead = 0 AND p.birthdate BETWEEN '1982-01-01' AND '1992-01-01'
AND enc.encounter_datetime BETWEEN '1996-01-01' AND '2012-12-31' AND answer.concept_id = 1205
GROUP BY obs.person_id
```

#### Final OpenClinica Simulated:

```
SELECT DISTINCT(ss.study_subject_id), s.unique_identifier, s.date_of_birth, s.gender, ((current_date -
s.date_of_birth)/365) as Age FROM subject s, study_subject ss, event_crf ec, item_data id,
item_form_metadata ifm, response_set rs WHERE s.subject_id = ss.subject_id AND ss.study_subject_id
= ec.study_subject_id AND id.item_id = ifm.item_id AND ifm.response_set_id = rs.response_set_id AND
id.event_crf_id = ec.event_crf_id AND s.date_of_birth BETWEEN '1982-01-01' AND '1992-01-01' AND
ec.date_interviewed BETWEEN '1996-01-01' AND '2012-12-31'
```

#### Final OpenClinica Manual:

```
SELECT DISTINCT(ss.study_subject_id), s.unique_identifier, s.date_of_birth, s.gender, ((current_date -
s.date_of_birth)/365) as Age FROM subject s, study_subject ss, event_crf ec, item_data id,
item_form_metadata ifm, response_set rs WHERE s.subject_id = ss.subject_id AND ss.study_subject_id
= ec.study_subject_id AND id.item_id = ifm.item_id AND ifm.response_set_id = rs.response_set_id AND
id.event_crf_id = ec.event_crf_id AND s.date_of_birth BETWEEN '1982-01-01' AND '1992-01-01' AND
ec.date_interviewed BETWEEN '1996-01-01' AND '2012-12-31'
```

### Scenario 3: Identifying adverse events, routine monitoring, side effects or contraindications caused by routine clinical care or clinical trial study protocols.

**Query 1: Find all patients of Physician Y who are on ART and have not had a clinical encounter in the past 3 months.**

#### Final OM Query Simulated:

```
SELECT distinct(obs.person_id) as ptid, pn.prefix, pn.given_name, pn.middle_name,
pn.family_name_prefix, pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender,
DATE_FORMAT(p.birthdate, '%d-%m-%Y') as birthdate, FLOOR(((DATEDIFF(CURDATE(),
p.birthdate))/365)) as Age, p.dead, pi.identifier, question.name as qn, answer.name as an,
DATE_FORMAT(obs.obs_datetime, '%d-%m-%Y') as birthdate from person p, person_name pn,
encounter enc, obs, concept_name question, concept_name answer, patient_identifier pi,
encounter_provider ep where p.person_id = pn.person_id and p.person_id = enc.patient_id and
p.person_id = obs.person_id and obs.concept_id = question.concept_id AND question.locale = 'en' AND
answer.locale = 'en' AND obs.value_coded = answer.concept_id AND obs.value_coded is not null AND
pi.patient_id = p.person_id AND question.concept_id = 161048 AND answer.concept_id in
(161037,161040,161042) AND ep.provider_id = 1 GROUP BY ptid
```

#### Final OM Query Manual:

```
SELECT distinct(obs.person_id) as ptid, pn.prefix, pn.given_name, pn.middle_name,
pn.family_name_prefix, pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender,
DATE_FORMAT(p.birthdate, '%d-%m-%Y') as birthdate, FLOOR(((DATEDIFF(CURDATE()),
p.birthdate))/365)) as Age, p.dead, pi.identifier, question.name as qn, answer.name as an,
DATE_FORMAT(obs.obs_datetime, '%d-%m-%Y') as birthdate from person p, person_name pn,
encounter enc, obs, concept_name question, concept_name answer, patient_identifier pi,
encounter_provider ep where p.person_id = pn.person_id and p.person_id = enc.patient_id and
p.person_id = obs.person_id and obs.concept_id = question.concept_id AND question.locale = 'en' AND
answer.locale = 'en' AND obs.value_coded = answer.concept_id AND obs.value_coded is not null AND
pi.patient_id = p.person_id AND question.concept_id = 161048 AND answer.concept_id in
(161037,161040,161042) AND ep.provider_id = 1 GROUP BY ptid
```

#### Final OC Query Simulated:

```
SELECT DISTINCT(ss.study_subject_id), s.unique_identifier, s.date_of_birth, s.gender, ((current_date -
s.date_of_birth)/365) as Age FROM subject s, study_subject ss, event_crf ec, study, item_data id,
item_form_metadata ifm, response_set rs WHERE s.subject_id = ss.subject_id AND ss.study_subject_id
= ec.study_subject_id AND id.item_id = ifm.item_id AND ifm.response_set_id = rs.response_set_id AND
id.event_crf_id = ec.event_crf_id AND ss.study_id = study.study_id
```

#### Final OC Query Manual:

```
SELECT DISTINCT(ss.study_subject_id), s.unique_identifier, s.date_of_birth, s.gender, ((current_date -
s.date_of_birth)/365) as Age FROM subject s, study_subject ss, event_crf ec, study, item_data id,
item_form_metadata ifm, response_set rs WHERE s.subject_id = ss.subject_id AND ss.study_subject_id
= ec.study_subject_id AND id.item_id = ifm.item_id AND ifm.response_set_id = rs.response_set_id AND
id.event_crf_id = ec.event_crf_id AND ss.study_id = study.study_id
```

#### Query 2: Find all patients of Clinic Z, or drug regimen ABC and DEF during time period G.

##### Final OMQuery:

```
SELECT distinct(obs.person_id) as ptid, pn.prefix, pn.given_name, pn.middle_name,
pn.family_name_prefix, pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender,
DATE_FORMAT(p.birthdate, '%d-%m-%Y') as birthdate, FLOOR(((DATEDIFF(CURDATE()),
p.birthdate))/365)) as Age, p.dead, pi.identifier, question.name as qn, answer.name as an,
DATE_FORMAT(obs.obs_datetime, '%d-%m-%Y') as birthdate from person p, person_name pn,
encounter enc, obs, concept_name question, concept_name answer, patient_identifier pi where
p.person_id = pn.person_id and p.person_id = enc.patient_id and p.person_id = obs.person_id and
obs.concept_id = question.concept_id AND question.locale = 'en' AND answer.locale = 'en' AND
obs.value_coded = answer.concept_id AND obs.value_coded is not null AND pi.patient_id = p.person_id
AND question.concept_id = 161048 AND answer.concept_id in (161037,161040,161042) AND
question.concept_id = 161048 AND answer.concept_id in (161037,161040,161042) AND
question.concept_id = 161048 AND answer.concept_id in (161037,161040,161042) AND
question.concept_id = 161048 AND answer.concept_id in (161037,161040,161042) AND
enc.encounter_datetime BETWEEN '2000-01-01' AND '2012-12-01' AND enc.location_id = 105 GROUP
BY ptid
```

##### Final OpenMRS Query Manual:

```
SELECT distinct(obs.person_id) as ptid, pn.prefix, pn.given_name, pn.middle_name,
pn.family_name_prefix, pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender,
DATE_FORMAT(p.birthdate, '%d-%m-%Y') as birthdate, FLOOR(((DATEDIFF(CURDATE()),
p.birthdate))/365)) as Age, p.dead, pi.identifier, question.name as qn, answer.name as
an, DATE_FORMAT(obs.obs_datetime, '%d-%m-%Y') as birthdate from person p, person_name pn,
encounter enc, obs, concept_name question, concept_name answer, patient_identifier pi where
p.person_id = pn.person_id and p.person_id = enc.patient_id and p.person_id = obs.person_id and
obs.concept_id = question.concept_id AND question.locale = 'en' AND answer.locale = 'en' AND
obs.value_coded = answer.concept_id AND obs.value_coded is not null AND pi.patient_id = p.person_id
AND question.concept_id = 161048 AND answer.concept_id in (161037,161040,161042)
```

AND enc.encounter\_datetime BETWEEN '2000-01-01' AND '2012-12-01' AND enc.location\_id = 105  
GROUP BY ptid

**Final OC Query:**

```
SELECT DISTINCT(ss.study_subject_id), s.unique_identifier, s.date_of_birth, s.gender, ((current_date -  
s.date_of_birth)/365) as Age FROM subject s, study_subject ss, event_crf ec, study, item_data id,  
item_form_metadata ifm, response_set rs WHERE s.subject_id = ss.subject_id AND ss.study_subject_id  
= ec.study_subject_id AND id.item_id = ifm.item_id AND ifm.response_set_id = rs.response_set_id AND  
id.event_crf_id = ec.event_crf_id AND ss.study_id = study.study_id AND ec.date_interviewed BETWEEN  
'2000-01-01' AND '2012-12-01' AND study.parent_study_id = 105
```

**Final OC Query Manual:**

```
SELECT DISTINCT(ss.study_subject_id), s.unique_identifier, s.date_of_birth, s.gender, ((current_date -  
s.date_of_birth)/365) as Age FROM subject s, study_subject ss, event_crf ec, study, item_data id,  
item_form_metadata ifm, response_set rs WHERE s.subject_id = ss.subject_id AND ss.study_subject_id  
= ec.study_subject_id AND id.item_id = ifm.item_id AND ifm.response_set_id = rs.response_set_id AND  
id.event_crf_id = ec.event_crf_id AND ss.study_id = study.study_id AND ec.date_interviewed BETWEEN  
'2000-01-01' AND '2012-12-01' AND study.parent_study_id = 105
```

## Appendix K. Manual and Generated SQL Queries by Scenario

### Real World Evaluation

**Scenario 1: Querying other provider's systems to identify prior medical records for a given patient.**

**Query 1: Find all encounters for Patient X**

**Final OpenMRS Simulated:**

```
SELECT enc.encounter_id, trim(date(enc.encounter_datetime)), et.name as EncType, loc.name as Clinic,
p.gender, p.birthdate, FLOOR(((DATEDIFF(CURDATE(), birthdate))/365)) as Age, pn.prefix,
pn.given_name, pn.middle_name, pn.family_name_prefix, pn.family_name, pn.family_name2,
pn.family_name_suffix, enc.patient_id FROM encounter enc JOIN encounter_type et ON
enc.encounter_type = et.encounter_type_id LEFT OUTER JOIN location loc ON loc.name =
enc.location_id JOIN person p ON p.person_id = enc.patient_id JOIN person_name pn ON pn.person_id
= enc.patient_id JOIN patient_identifier pi ON pi.patient_id = pn.person_id WHERE pi.identifier = '██████████'
AND p.birthdate = '██████████' and pn.given_name = '██' and pn.middle_name = '██' and pn.family_name
= '██' and p.gender = '██' and ORDER BY enc.encounter_datetime desc
```

**Final OpenMRS Manual:**

```
SELECT pi.identifier, pn.given_name, pn.middle_name, pn.family_name_prefix, pn.family_name,
pn.family_name2, pn.family_name_suffix, pn.family_name_suffix, pn.degree, p.gender, p.birthdate,
FLOOR(((DATEDIFF(CURDATE(), p.birthdate))/365)) as Age, enc.encounter_id, et.name FROM person
p, person_name pn, patient_identifier pi, encounter enc, encounter_type et WHERE pn.person_id =
p.person_id AND pi.identifier = '██████████' AND p.birthdate = '██████████' and pn.given_name = '██' and
pn.middle_name = '██' and pn.family_name = '██' and p.gender = '██' and pi.patient_id = p.person_id
AND enc.patient_id = p.person_id AND enc.encounter_type = et.encounter_type_id
```

**Final OpenClinica Simulated:**

```
SELECT s.subject_id, s.unique_identifier, ss.study_subject_id, se.study_event_id, se.location,
date(se.date_start), sed.name, sed.type FROM subject s, study_subject ss, study_event se,
study_event_definition sed WHERE ss.study_subject_id = '██████████' AND s.subject_id = ss.study_subject_id
AND ss.study_subject_id = se.study_subject_id AND se.study_event_definition_id =
sed.study_event_definition_id ORDER BY date
```

**Final OpenClinica Manual:**

```
SELECT s.subject_id, s.unique_identifier, ss.study_subject_id, se.study_event_id, se.location,
date(se.date_start), sed.name, sed.type FROM subject s, study_subject ss, study_event se,
study_event_definition sed WHERE ss.study_subject_id = '██████████' AND s.subject_id = ss.study_subject_id
AND ss.study_subject_id = se.study_subject_id AND se.study_event_definition_id =
sed.study_event_definition_id ORDER BY date
```

**Query 2: Find all future visit dates for Patient X**

**Final OpenMRS Simulated:**

```
SELECT enc.encounter_datetime, pn.prefix, pn.given_name, pn.middle_name, pn.family_name_prefix,
pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender, p.birthdate,
FLOOR(((DATEDIFF(CURDATE(), p.birthdate))/365)) as Age, p.dead, pi.identifier FROM person p,
person_name pn, patient_identifier pi, encounter enc WHERE p.person_id = pn.person_id and
p.person_id = pi.patient_id and enc.patient_id = p.person_id AND pi.identifier = '██████████' AND
pn.given_name = '██' AND pn.middle_name = '██' AND pn.family_name = '██' AND enc.encounter_datetime
>='2010-01-11' AND p.gender = '██' AND p.birthdate = '██████████'
```

**Final OpenMRS Manual:**

```
SELECT enc.encounter_datetime, pn.prefix, pn.given_name, pn.middle_name, pn.family_name_prefix,
pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender, p.birthdate,
FLOOR(((DATEDIFF(CURDATE(), p.birthdate))/365)) as Age, p.dead, pi.identifier FROM person p,
person_name pn, patient_identifier pi, encounter enc WHERE p.person_id = pn.person_id and
p.person_id = pi.patient_id and enc.patient_id = p.person_id AND pi.identifier = ██████████ AND
pn.given_name = ██████ AND pn.middle_name = ██████ AND pn.family_name = 'Y' AND enc.encounter_datetime
>= '2010-01-11' AND p.gender = ██████ AND p.birthdate = ██████████
```

### Query 3: Identify recent laboratory data for Patient X

#### Final OpenMRS Simulated:

```
SELECT distinct(pi.identifier) as ptid, pn.prefix, pn.given_name, pn.middle_name, pn.family_name_prefix,
pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender, DATE_FORMAT(p.birthdate, '%d-
%m-%Y') as birthdate, FLOOR(((DATEDIFF(CURDATE(), p.birthdate))/365)) as Age, p.dead,
question.concept_id, question.name, obs.value_numeric, answer.name FROM concept_name question,
obs, concept c, concept_name answer, person p, person_name pn, patient_identifier pi WHERE
obs.concept_id = question.concept_id and answer.concept_id = obs.concept_id and obs.concept_id =
c.concept_id AND question.locale = 'en' AND answer.locale = 'en' and p.person_id = pn.person_id and
pi.patient_id = p.person_id and obs.person_id = p.person_id and c.class_id in (1,2) AND p.dead = 0 AND
p.gender = ██████ AND pn.given_name = ██████ AND pn.middle_name = ██████ AND pn.family_name = ██████ AND
p.birthdate = ██████████ AND pi.identifier = ██████████ AND question.concept_id in (161024, 161026,
161035) AND obs.obs_datetime BETWEEN '2005-05-05' AND '2010-05-05' AND question.concept_id in
(161024, 161026, 161035) AND obs.obs_datetime BETWEEN '2005-05-05' AND '2010-05-05' AND
question.concept_id in (161024, 161026, 161035) AND obs.obs_datetime BETWEEN '2005-05-05' AND
'2010-05-05' AND question.concept_id in (161024, 161026, 161035) AND obs.obs_datetime BETWEEN
'2005-05-05' AND '2010-05-05' GROUP BY question.concept_id, question.name, obs.value_numeric
ORDER BY obs.obs_datetime
```

#### Final OpenMRS Manual:

```
SELECT distinct(pi.identifier) as ptid, pn.prefix, pn.given_name, pn.middle_name, pn.family_name_prefix,
pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender, DATE_FORMAT(p.birthdate, '%d-
%m-%Y') as birthdate, FLOOR(((DATEDIFF(CURDATE(), p.birthdate))/365)) as Age, p.dead,
question.concept_id, question.name, obs.value_numeric, answer.name FROM concept_name question,
obs, concept c, concept_name answer, person p, person_name pn, patient_identifier pi WHERE
obs.concept_id = question.concept_id and answer.concept_id = obs.concept_id and obs.concept_id =
c.concept_id AND question.locale = 'en' AND answer.locale = 'en' and p.person_id = pn.person_id and
pi.patient_id = p.person_id and obs.person_id = p.person_id and c.class_id in (1,2) AND p.dead = 0 AND
p.gender = ██████ AND pn.given_name = ██████ AND pn.middle_name = ██████ AND pn.family_name = ██████ AND
p.birthdate = ██████████ AND pi.identifier = ██████████ AND question.concept_id in (161024, 161026,
161035) AND obs.obs_datetime BETWEEN '2005-05-05' AND '2010-05-05' GROUP BY
question.concept_id, question.name, obs.value_numeric ORDER BY obs.obs_datetime
```

### Scenario 2: Determining the inclusion criteria for clinical trials, and querying systems to determine patients who will be included or excluded from the trial

#### Final OpenMRS Simulated:

```
SELECT p.person_id as ptid, pn.prefix, pn.given_name, pn.middle_name, pn.family_name_prefix,
pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender, p.birthdate,
FLOOR(((DATEDIFF(CURDATE(), p.birthdate))/365)) as Age, p.dead, pi.identifier FROM person p,
person_name pn, concept_name question, encounter enc, patient_identifier pi, obs LEFT OUTER JOIN
concept_name answer ON (answer.concept_id = obs.value_coded) WHERE obs.concept_id =
question.concept_id AND obs.person_id = p.person_id AND p.person_id = pn.person_id AND obs.voided
!= 1 AND answer.locale = 'en' AND question.locale = 'en' AND enc.encounter_id = obs.encounter_id AND
pi.patient_id = p.person_id AND p.dead = 0 AND p.birthdate BETWEEN '1982-01-01' AND '1992-01-01'
AND question.concept_id = (161034) AND answer.concept_id = 1065 AND obs.obs_datetime BETWEEN
'1990-01-01' AND '2011-12-31' AND obs.obs_datetime BETWEEN '1990-01-01' AND '2011-12-31' AND
obs.obs_datetime BETWEEN '1990-01-01' AND '2011-12-31' AND obs.obs_datetime BETWEEN '1990-
```

```
01-01' AND '2011-12-31' AND obs.obs_datetime BETWEEN '1990-01-01' AND '2011-12-31' AND
obs.obs_datetime BETWEEN '1990-01-01' AND '2011-12-31' AND obs.obs_datetime BETWEEN '1990-
01-01' AND '2011-12-31' AND obs.obs_datetime BETWEEN '1990-01-01' AND '2011-12-31' AND
obs.obs_datetime BETWEEN '1990-01-01' AND '2011-12-31' AND obs.obs_datetime BETWEEN '1990-
01-01' AND '2011-12-31' GROUP BY obs.person_id
```

#### Final OpenMRS Manual:

```
SELECT p.person_id as ptid, pn.prefix, pn.given_name, pn.middle_name, pn.family_name_prefix,
pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender, p.birthdate,
FLOOR(((DATEDIFF(CURDATE(), p.birthdate))/365)) as Age, p.dead, pi.identifier FROM person p,
person_name pn, concept_name question, encounter enc, patient_identifier pi, obs LEFT OUTER JOIN
concept_name answer ON (answer.concept_id = obs.value_coded) WHERE obs.concept_id =
question.concept_id AND obs.person_id = p.person_id AND p.person_id = pn.person_id AND obs.voided
!= 1 AND answer.locale = 'en' AND question.locale = 'en' AND enc.encounter_id = obs.encounter_id AND
pi.patient_id = p.person_id AND p.dead = 0 AND p.birthdate BETWEEN '1982-01-01' AND '1992-01-
01' AND question.concept_id = (161034) AND answer.concept_id = 1065 AND obs.obs_datetime
BETWEEN '1990-01-01' AND '2011-12-31' GROUP BY obs.person_id
```

#### Final OpenClinica Simulated:

```
SELECT DISTINCT(ss.study_subject_id), s.unique_identifier, s.date_of_birth, s.gender, ((current_date -
s.date_of_birth)/365) as Age FROM subject s, study_subject ss, event_crf ec, item_data id,
item_form_metadata ifm, response_set rs WHERE s.subject_id = ss.subject_id AND ss.study_subject_id
= ec.study_subject_id AND id.item_id = ifm.item_id AND ifm.response_set_id = rs.response_set_id AND
id.event_crf_id = ec.event_crf_id AND s.date_of_birth BETWEEN '1982-01-01' AND '1992-01-01'
```

#### Final OpenClinica Manual:

```
SELECT DISTINCT(ss.study_subject_id), s.unique_identifier, s.date_of_birth, s.gender, ((current_date -
s.date_of_birth)/365) as Age FROM subject s, study_subject ss, event_crf ec, item_data id,
item_form_metadata ifm, response_set rs WHERE s.subject_id = ss.subject_id AND ss.study_subject_id
= ec.study_subject_id AND id.item_id = ifm.item_id AND ifm.response_set_id = rs.response_set_id AND
id.event_crf_id = ec.event_crf_id AND s.date_of_birth BETWEEN '1982-01-01' AND '1992-01-01'
```

### Scenario 3: Identifying adverse events, routine monitoring, side effects or contraindications caused by routine clinical care or clinical trial study protocols.

**Query 1: Find all patients of Physician Y who are on ART and have not had a clinical encounter in the past 3 months.**

#### Final OpenMRS Simulated:

```
SELECT distinct(obs.person_id) as ptid, pn.prefix, pn.given_name, pn.middle_name,
pn.family_name_prefix, pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender,
DATE_FORMAT(p.birthdate, '%d-%m-%Y') as birthdate, FLOOR(((DATEDIFF(CURDATE(),
p.birthdate))/365)) as Age, p.dead, pi.identifier, question.name as qn, answer.name as an,
DATE_FORMAT(obs.obs_datetime, '%d-%m-%Y') as birthdate from person p, person_name pn,
encounter enc, obs, concept_name question, concept_name answer, patient_identifier pi,
encounter_provider ep where p.person_id = pn.person_id and p.person_id = enc.patient_id and
p.person_id = obs.person_id and obs.concept_id = question.concept_id AND question.locale = 'en' AND
answer.locale = 'en' AND obs.value_coded = answer.concept_id AND obs.value_coded is not null AND
pi.patient_id = p.person_id AND AND answer.concept_id in (161042) AND obs.obs_datetime BETWEEN
'2005-05-05' AND '2009-05-05' AND AND answer.concept_id in (161042) AND obs.obs_datetime
BETWEEN '2005-05-05' AND '2009-05-05' AND AND answer.concept_id in (161042) AND
obs.obs_datetime BETWEEN '2005-05-05' AND '2009-05-05' AND AND answer.concept_id in (161042)
AND obs.obs_datetime BETWEEN '2005-05-05' AND '2009-05-05' AND ep.provider_id = 1 GROUP BY
ptid
```

#### Final OpenMRS Manual:

```
SELECT distinct(obs.person_id) as ptid, pn.prefix, pn.given_name, pn.middle_name,
pn.family_name_prefix, pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender,
DATE_FORMAT(p.birthdate, '%d-%m-%Y') as birthdate, FLOOR(((DATEDIFF(CURDATE()),
p.birthdate))/365)) as Age, p.dead, pi.identifier, question.name as qn, answer.name as
an, DATE_FORMAT(obs.obs_datetime, '%d-%m-%Y') as birthdate from person p, person_name pn,
encounter enc, obs, concept_name question, concept_name answer, patient_identifier pi,
encounter_provider ep where p.person_id = pn.person_id and p.person_id = enc.patient_id and
p.person_id = obs.person_id and obs.concept_id = question.concept_id AND question.locale = 'en' AND
answer.locale = 'en' AND obs.value_coded = answer.concept_id AND obs.value_coded is not null AND
pi.patient_id = p.person_id and answer.concept_id = 161042 AND obs.obs_datetime BETWEEN '2005-
05-05' AND '2009-05-05' AND ep.provider_id = 1 GROUP BY ptid
```

#### **Final OpenClinica Simulated:**

```
SELECT DISTINCT(ss.study_subject_id), s.unique_identifier, s.date_of_birth, s.gender, ((current_date -
s.date_of_birth)/365) as Age FROM subject s, study_subject ss, event_crf ec, study, item_data id,
item_form_metadata ifm, response_set rs WHERE s.subject_id = ss.subject_id AND ss.study_subject_id
= ec.study_subject_id AND id.item_id = ifm.item_id AND ifm.response_set_id = rs.response_set_id AND
id.event_crf_id = ec.event_crf_id AND ss.study_id = study.study_id
```

#### **Final OpenClinica Manual:**

```
SELECT DISTINCT(ss.study_subject_id), s.unique_identifier, s.date_of_birth, s.gender, ((current_date -
s.date_of_birth)/365) as Age FROM subject s, study_subject ss, event_crf ec, study, item_data id,
item_form_metadata ifm, response_set rs WHERE s.subject_id = ss.subject_id AND ss.study_subject_id
= ec.study_subject_id AND id.item_id = ifm.item_id AND ifm.response_set_id = rs.response_set_id AND
id.event_crf_id = ec.event_crf_id AND ss.study_id = study.study_id
```

#### **Query 2: Find all patients of Clinic Z, or drug regimen ABC and DEF during time period G.**

##### **Final OpenMRS Simulated:**

```
SELECT distinct(obs.person_id) as ptid, pn.prefix, pn.given_name, pn.middle_name,
pn.family_name_prefix, pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender,
DATE_FORMAT(p.birthdate, '%d-%m-%Y') as birthdate, FLOOR(((DATEDIFF(CURDATE()),
p.birthdate))/365)) as Age, p.dead, pi.identifier, question.name as qn, answer.name as an,
DATE_FORMAT(obs.obs_datetime, '%d-%m-%Y') as birthdate from person p, person_name pn,
encounter enc, obs, concept_name question, concept_name answer, patient_identifier pi,
encounter_provider ep where p.person_id = pn.person_id and p.person_id = enc.patient_id and
p.person_id = obs.person_id and obs.concept_id = question.concept_id AND question.locale = 'en' AND
answer.locale = 'en' AND obs.value_coded = answer.concept_id AND obs.value_coded is not null AND
pi.patient_id = p.person_id AND p.dead = 0 AND answer.concept_id in (86663) AND answer.concept_id
in (86663) AND question.concept_id = 161048 AND answer.concept_id in (86663) AND
answer.concept_id in (86663)
```

##### **Final OpenMRS Manual:**

```
SELECT distinct(obs.person_id) as ptid, pn.prefix, pn.given_name, pn.middle_name,
pn.family_name_prefix, pn.family_name, pn.family_name2, pn.family_name_suffix, p.gender,
DATE_FORMAT(p.birthdate, '%d-%m-%Y') as birthdate, FLOOR(((DATEDIFF(CURDATE()),
p.birthdate))/365)) as Age, p.dead, pi.identifier, question.name as qn, answer.name as
an, DATE_FORMAT(obs.obs_datetime, '%d-%m-%Y') as birthdate from person p, person_name pn,
encounter enc, obs, concept_name question, concept_name answer, patient_identifier pi,
encounter_provider ep where p.person_id = pn.person_id and p.person_id = enc.patient_id and
p.person_id = obs.person_id and obs.concept_id = question.concept_id AND question.locale = 'en' AND
answer.locale = 'en' AND obs.value_coded = answer.concept_id AND obs.value_coded is not null AND
pi.patient_id = p.person_id AND p.dead = 0 AND answer.concept_id = 86663
```

##### **Final OpenClinica Simulated:**

```
SELECT DISTINCT(ss.study_subject_id), s.unique_identifier, s.date_of_birth, s.gender, ((current_date -
```

```
s.date_of_birth)/365) as Age FROM subject s, study_subject ss, event_crf ec, study, item_data id,
item_form_metadata ifm, response_set rs WHERE s.subject_id = ss.subject_id AND ss.study_subject_id
= ec.study_subject_id AND id.item_id = ifm.item_id AND ifm.response_set_id = rs.response_set_id AND
id.event_crf_id = ec.event_crf_id AND ss.study_id = study.study_id AND ec.date_interviewed BETWEEN
'1983-07-01' AND '2001-12-01' AND study.parent_study_id = 3
```

**Final OpenClinica Manual:**

```
SELECT DISTINCT(ss.study_subject_id), s.unique_identifier, s.date_of_birth, s.gender, ((current_date -
s.date_of_birth)/365) as Age FROM subject s, study_subject ss, event_crf ec, study, item_data id,
item_form_metadata ifm, response_set rs WHERE s.subject_id = ss.subject_id AND ss.study_subject_id
= ec.study_subject_id AND id.item_id = ifm.item_id AND ifm.response_set_id = rs.response_set_id AND
id.event_crf_id = ec.event_crf_id AND ss.study_id = study.study_id AND ec.date_interviewed BETWEEN
'1983-07-01' AND '2001-12-01' AND study.parent_study_id = 3
```