UNIVERSITY OF TROMSØ UIT

FACULTY OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF COMPUTER SCIENCE

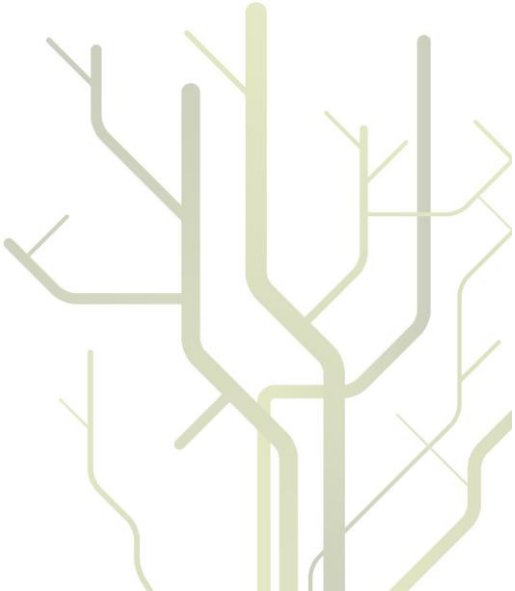
# Snow Integrated Communicable Disease Prediction Service

**Kassaye Yitbarek Yigzaw**

INF-3997

Master's Thesis in Telemedicine and E-health

June 2012



UNIVERSITY OF TROMSØ UIT

FACULTY OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF COMPUTER SCIENCE

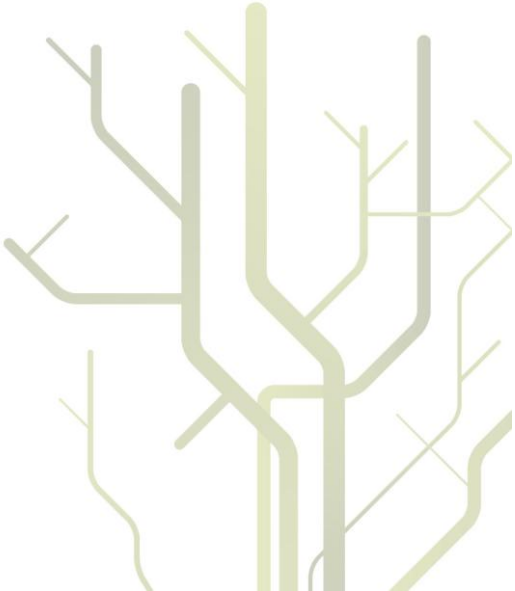
# Snow Integrated Communicable Disease Prediction Service

**Kassaye Yitbarek Yigzaw**

INF-3997

Master's Thesis in Telemedicine and E-health

June 2012



## Abstract

**Objective:** This thesis mainly focused on construction of an integrated infectious disease prediction service that predicts and visualizes prediction results in time and space.

**Methods:** We have used weekly aggregated laboratory confirmed cases of various diseases collected from the Snow system, which is an infectious disease surveillance system that covers Troms and Finnmark counties of north Norway. Influenza A dataset is applied for modeling SIR(S) model and various diseases datasets applied to a Bayesian model.

The infectious disease prediction service prototype was constructed following an iterative and incremental approach where the entire development process was composed of four activities.

**Results:** The prediction service framework facilitates the process of integrating various models and allows their evaluation. Currently, the system contains two mathematical models that demonstrate the effectiveness of the architecture in integrating new models.

**Conclusion:** The framework can significantly improve the status of disease prediction systems, investment and time of development. It also speeds up mathematical modeling through its integrated environment for testing and evaluating different mathematical models against other existing models. Thus, the project contributes to improve the overall disease prediction accuracy and increase the benefits from prediction.



**Keywords:** Infectious disease, Influenza, Mathematical model, Prediction, Mathematical model evaluation, Spatiotemporal Epidemiological Modeler, Visualization, Integrated infectious disease prediction.

# Preface

Last summer I had a chance to take part in IBM Extreme Blue internship. During the internship I have worked on real-time disease surveillance system project, where I got to learn about an infectious diseases outbreak detection model, C-SiZer (Skrøvseth et al. 2012). This is how my attachment with epidemiological models started. Later during a discussion with Johan Gustav Bellika, he pointed me to a couple of topics for my master thesis and I chose Infectious disease prediction service.

The current infectious diseases threats, both naturally occurring and caused by bioterrorism attacks, raised major urgent concerns with regard to public health preparedness and decision making. The objective of the thesis is creation of an integrated infectious disease prediction service that make spatio-temporal predictions and visualize.

This thesis is part of the Snow project, which is an on-going applied research project at Norwegian Centre for Telemedicine (NST) and Tromsø Telemedicine Laboratory (TTL). The Snow project is mainly focused on creation of computer systems for communicable disease prediction, detection, and control. The infectious diseases surveillance data (Bellika et al. 2009) and Bayesian prediction model (Geilhufe et al. 2012) used in the thesis are also part of the Snow project.

In the process of developing this thesis there are several individuals and institutions whose contributions were incredible and without them I could not have completed this thesis. First, I would like to express my sincere gratitude to my supervisor Johan Gustav Bellika for giving me a chance to work with him. His guidance, motivation and enthusiasm helped me in all the time of the masters program. I also thank him for encouraging me to solve problems by myself, it has stretched me as a researcher. I also thank his display of concern over my personal development and career growth.

I am deeply indebted to my co-supervisor, Marc Geilhufe, whose prediction model has

been used in the thesis. He has filled me with knowledge on his model and mathematical modeling in general. He has always been there when I needed a discussion and shared valuable insights.

My sincere thanks also go to my co-supervisor, Gunnar Hartvigsen, for his guidance and valuable comments from his wealth of experience. I am thankful for his kindness lending me books relevant to my work.

My sincere thanks also go to Stein Olav Skrøvseth for sharing his valuable time for discussions on the prediction models and requirement gathering. He has always been open for discussion and shared valuable insights.

I would like to thank NST and TTL for providing me office during the thesis period. I also appreciate all the kind helps from the Snow project team, especially Lars Ilebrekke for providing me the research data from the Snow system and discussion about the Snow system architecture. I am also thankful to Gunnar Skov Simonsen for allowing the microbiology lab data to be used in the Snow system.

This thesis has used an open source software package called Spatiotemporal Epidemiological Modeler (STEM). I am deeply indebted to Eclipse Foundation and STEM development team for making the software freely available. I am also very grateful for all the support and advice from James Kaufman and Stefan Edlund, in the development of Norway map plug-ins for STEM and Influenza A modeling.

I am thankful to Gro Berntsen for her valuable discussion in the requirement gathering and infectious disease epidemiology.

I am also thankful to the Norwegian State Educational Loan Fund, Lånekassen, for the financial support throughout my masters study period.

I am also thankful to IBM for offering me the opportunity to work in the Extreme Blue internship. I would like to thank all the people who made the internship a success, including Jan Fredrik Sagdahl, François Commagnac, Johan Gustav Bellika, Stein Olav Skrøvseth, Gunnar Hartvigsen, Jacob Eisinger, Vincent Tassy and the internship students.

I am grateful to my Creator and Savior, almighty God for being my source of strength. Last but not least, I would like to thank my family and friends for their support in any respect during the masters period.

# Contents

**Preface** **ii**

**List of Figures** **x**

**List of Tables** **xii**

**Abbreviations** **xiv**

**1 Introduction** **2**

1.1 Background and Motivation . . . . . 2

1.2 Research problems . . . . . 3

1.3 Materials and Methods . . . . . 3

1.4 Project Contribution . . . . . 4

1.5 Organization of the Thesis . . . . . 5

**2 Theoretical Framework** **8**

2.1 Introduction . . . . . 8

2.2 Terminology . . . . . 8

2.3 Infectious Diseases . . . . . 10

2.4 Mathematical Models . . . . . 11

2.4.1 History of Mathematical Epidemiological Models . . . . . 11

2.4.2 Compartmental Models . . . . . 12

2.4.3 Bayesian Models . . . . . 15

2.4.4 Mathematical Models Comparison . . . . . 18

2.5 Infectious Disease Prediction and Detection Systems . . . . . 19

2.5.1 Infectious Disease Detection Systems . . . . . 19

2.5.2 Infectious Disease Prediction Tools . . . . . 20

2.5.3 Spatiotemporal Epidemiological Modeler . . . . . 21

2.6 The Snow Agent System . . . . . 25
2.7 Visualization . . . . . 28
2.8 Summary . . . . . 31

**3 Materials and Methods** **34**
3.1 Introduction . . . . . 34
3.2 Materials . . . . . 34
3.2.1 Hardware and Software . . . . . 34
3.2.2 Study Area and Data Source . . . . . 35
3.2.3 Data Analysis . . . . . 35
3.3 Software Development . . . . . 35
3.3.1 User-Centred Design . . . . . 36
3.3.2 Requirement Specification . . . . . 36
3.4 Mathematical Models . . . . . 36
3.5 Critique of the Methods Used . . . . . 37
3.6 Summary . . . . . 37

**4 Software Requirements Specification** **38**
4.1 Introduction . . . . . 38
4.2 System Description . . . . . 38
4.3 Requirements Process and Specification Method . . . . . 39
4.4 Source of Requirements . . . . . 40
4.5 Functional requirements . . . . . 42
4.6 Use Case . . . . . 45
4.7 Non-functional requirements . . . . . 48
4.7.1 Scalability . . . . . 48
4.7.2 Extensibility . . . . . 48
4.7.3 Usability . . . . . 49
4.8 Summary . . . . . 49

**5 Design** **50**
5.1 Introduction . . . . . 50
5.2 Design Considerations . . . . . 50
5.3 Architectural Design . . . . . 51
5.4 Data Design . . . . . 53
5.5 Components Design . . . . . 54

5.6 Interface Design . . . . . 64
5.7 Summary . . . . . 66

**6 Implementation and Testing** **68**
6.1 Introduction . . . . . 68
6.2 Programming language and Technologies . . . . . 68
6.2.1 Web Services . . . . . 69
6.2.2 Object-Relational Mapping . . . . . 69
6.3 Data Layer . . . . . 69
6.4 Business Layer . . . . . 72
6.5 Presentation Layer . . . . . 74
6.6 Testing . . . . . 75
6.7 Requirements Matrix . . . . . 75
6.8 Summary . . . . . 76

**7 Mathematical Models and Evaluation** **78**
7.1 Introduction . . . . . 78
7.2 Influenza A SIR(S) Model . . . . . 78
7.3 Bayesian Model . . . . . 80
7.4 Results and Discussion . . . . . 81

**8 Results and Discussion** **86**
8.1 Introduction . . . . . 86
8.2 Prediction Service Framework . . . . . 86
8.2.1 Data Source . . . . . 87
8.2.2 Prediction . . . . . 87
8.2.3 Visualization . . . . . 87
8.3 Evaluation of Mathematical Models . . . . . 88
8.3.1 Bayesian Model . . . . . 88
8.3.2 Compartmental Model . . . . . 89
8.4 Importance of the Prediction Service . . . . . 89
8.5 Comparison with Similar Studies . . . . . 90
8.6 Limitations . . . . . 91

**9 Conclusion and Future Work** **92**
9.1 Conclusion . . . . . 92
9.2 Future Work . . . . . 93

**Bibliography** **95**

**A Prediction result xml schema definition** **108**

**B Prediction schedule xml schema definition** **109**

**C Municipalities of Troms and Finnmark counties** **110**

**D Bayesian Model Weekly and Monthly Predictions** **113**

# List of Figures


2.1 Infection-disease evolution, Source: Figure 2.1 in (Ramirez 2008) . . . . 10
2.2 A simple SIR model . . . . . 13
2.3 SIR(S) model, Source: (Edlund et al. 2011a) . . . . . 15
2.4 The Snow Agent System, Source: (Bellika et al. 2007) . . . . . 26
2.5 Snow Agent System laboratory data extraction from UNN . . . . . 27
2.6 A fragment of Snow XML report . . . . . 28
2.7 John Snow map for describing the Broad Street pump cholera outbreak of 1854 (Frerichs 2006) . . . . . 30
2.8 Screenshot of HealthMap, Source (CDC 2011*a*) . . . . . 31

4.1 Use case diagram for the Infectious disease Prediction Service Framework 46

5.1 Infectious Disease Prediction Service architecture . . . . . 52
5.2 Snow Interface Class Diagram . . . . . 55
5.3 Snow Interface Sequence Diagram . . . . . 55
5.4 Database Access Class Diagram . . . . . 56
5.5 Insert Prediction Schedule Sequence Diagram . . . . . 56
5.6 Prediction Manager Class Diagram . . . . . 57
5.7 Prediction Manager Sequence Diagram . . . . . 58
5.8 STEM Interface module Class Diagram . . . . . 59
5.9 STEM Interface Sequence Diagram . . . . . 59
5.10 Bayesian Model Interface Class Diagram . . . . . 60
5.11 Bayesian Model Interface Sequence Diagram . . . . . 61
5.12 Visualization Manager Class Diagram . . . . . 61
5.13 Visualization Manager Sequence Diagram . . . . . 62
5.14 Comparison Manager Class Diagram . . . . . 62
5.15 Error Function Class Diagram . . . . . 63

5.16 Error Function Sequence Diagram . . . . . 64
5.17 Screenshot of Main interface . . . . . 65
5.18 Screenshot of Comparison interface . . . . . 65
5.19 Screenshot of Schedule interface . . . . . 66
5.20 Screenshot of Data Upload interface . . . . . 66

6.1 JAX-RS resource class code fragment for querying schedule and prediction result . . . . . 70
6.2 JPA code fragment for inserting new schedule into database . . . . . 71
6.3 A csv file fragment processed by the Snow interface (Note: municipalities are defined using ISO 3166-2 code) . . . . . 71
6.4 Sample prediction schedule . . . . . 72
6.5 Code fragment to deserialize the schedule retrieved from database . . . 72
6.6 A fragment of sample prediction result in database . . . . . 73
6.7 Code fragment of a method that implements NRMSE . . . . . 74

7.1 Weekly Influenza A cases (summed over all locations) for fitted model (red) and actual Influenza A data (blue) . . . . . 82
7.2 Weekly Influenza A cases (summed over all locations) for actual Influenza A (blue), Bayesian model (red) and SIR(S) model (green) . . . 83
7.3 Monthly Influenza A cases (summed over all locations) for actual Influenza A (blue), Bayesian (red) and SIR(S) (green) models . . . . . 83

A.1 Prediction result xml schema definition . . . . . 108

B.1 Prediction schedule xml schema definition . . . . . 109

D.1 Weekly RS-virus cases (summed over all locations) for actual RS-virus (blue) and Bayesian model (red) . . . . . 113
D.2 Monthly RS-virus cases (summed over all locations) for actual RS-virus (blue) and Bayesian model (red) . . . . . 114
D.3 Weekly Rhinovirus cases (summed over all locations) for actual Rhinovirus (blue) and Bayesian model (red) . . . . . 114
D.4 Monthly Rhinovirus cases (summed over all locations) for actual Rhinovirus (blue) and Bayesian model (red) . . . . . 115
D.5 Weekly Norovirus cases (summed over all locations) for actual Norovirus (blue) and Bayesian model (red) . . . . . 115

D.6 Monthly Norovirus cases (summed over all locations) for actual Norovirus (blue) and Bayesian model (red) . . . . . 116
D.7 Weekly Mycoplasma pneumoniae cases (summed over all locations) for actual Mycoplasma pneumoniae (blue) and Bayesian model (red) . . . 116
D.8 Monthly Mycoplasma pneumoniae cases (summed over all locations) for actual Mycoplasma pneumoniae (blue) and Bayesian model (red) . . . 117

# List of Tables


2.1 Schema of the Snow database reports table . . . . . 28

3.1 Software and hardware materials . . . . . 34

4.1 Persona of Epidemiologist . . . . . 41
4.2 Persona of Mathematician . . . . . 41
4.3 Persona of General Practitioner . . . . . 42
4.4 Persona of Public . . . . . 42
4.5 Functional Requirement one . . . . . 43
4.6 Functional Requirement two . . . . . 43
4.7 Functional Requirement three . . . . . 43
4.8 Functional Requirement four . . . . . 44
4.9 Functional Requirement five . . . . . 44
4.10 Functional Requirement six . . . . . 44
4.11 Functional Requirement seven . . . . . 45

6.1 Requirement matrix . . . . . 75

7.1 NRMSE values of Bayesian model weekly and monthly (defined as 4 weeks) predictions of various diseases . . . . . 81

C.1 Municipalities of Troms county and ISO-code . . . . . 111
C.2 Municipalities of Finnmark county and ISO-code . . . . . 112

# Abbreviations

| | |
|---|---|
| AgD | Agent Daemon |
| AJAX | Asynchronous JavaScript and XML |
| CSV | Comma Separated Value |
| DSS | Decision Support Systems |
| EHR | Electronic Health Record system |
| EMF | Eclipse Modeling Framework |
| GP | General Practitioner |
| IBM | International Business Machines |
| INLA | Integrated Nested Laplace Approximations |
| JAX-RS | Java API for RESTful Web Services |
| JPA | Java Persistence API |
| MC | Mission Controller |
| MCMC | Markov Chain Monte Carlo |
| NRMSE | Normalized Root Mean Square Error |
| OHF | Open Healthcare Framework |
| SAS | Snow Agent Server |
| STEM | Spatiotemporal Epidemiological Modeler |

# Chapter 1

# Introduction

## 1.1 Background and Motivation

The current infectious diseases threats, both naturally occurring and caused by bioterrorism attacks, such as H1N1 (Dawood et al. 2009), SARS (Massad et al. 2005), and 2001 anthrax attacks (Fong & Alibek 2009) (Chang et al. 2003), raised major urgent concerns with regard to public health preparedness and decision making.

Mathematical models have been utilized in analyzing how infectious diseases will spread and its effective control mechanisms which significantly improves outbreak prevention and control capabilities by providing a lead-time to allow governments and health-care services to respond to outbreaks in a timely fashion (Myers et al. 2000). Short-term predictions could also be important in daily clinical care and for the public.

This thesis aims to construct an infectious diseases prediction service that predicts the spatio-temporal progression of communicable diseases in the same manner as weather forecasts. The service is planned to be a component of the Snow system (Bellika et al. 2007), which is an infectious disease surveillance system in northern Norway. A recent review (Cheng et al. 2009) on national influenza surveillance websites reported no websites attempted to quantitatively predict influenza. The prediction component could make the Snow system one of the first of this kind.

Infectious diseases have different dynamic of disease spread all requiring potentially different datasets, and models for their prediction (Kaufman et al. 2008). As a result,

the prediction environment needs to scale to integrate new mathematical models.

## 1.2 Research problems

This research aims to address the following research questions:

*1. How can we construct a generic infectious disease prediction service framework that enables integration of new mathematical models?*

Mathematical models simplify the dynamics of infectious diseases spread in a way that sufficiently decreases complexity; as a result no model will ever be completely accurate (Coiera 2003). Model selection and assessment of performance is an important part of any analysis and, indeed, is central to the pursuit of science in general (Kadane & Lazar 2004). This led us to a research sub-question,

*1.1. How can we assess performances of the models in the system?*

*2. How can we visualize prediction results in a way that facilitates access to prediction results and support decision making?*

Visualization can support decision makers by providing important information in an intuitively understandable way. Studies have been examining information visualizations effect on decision accuracy, but it is not yet well understood (Burstein et al. 2008).

## 1.3 Materials and Methods

In this thesis we have used software tools (i.e. Eclipse, Spatiotemporal Epidemiological Modeler (STEM), and R (Venables & Smith 2012), computers and infectious diseases laboratory data collected from a disease surveillance system.

An infectious disease surveillance system prototype was developed using a modified engineering method described in (Denning et al. 1989). The method follows an iterative and incremental approach where the entire development process composed of activities such as:

1. State requirements;
2. State specifications;
3. Design and implement the system;
4. Test the system.

The requirements analysis, specification and design architecture are defined using waterfall approach, followed by iterative and incremental implementation of sub systems.

We built a seasonally modulated SIR(S) Influenza A model with air transportation model (Edlund et al. 2011b) using STEM .

The predictions of SIR(S) and Bayesian models are compared against two years unseen laboratory data using Normalized Root Mean Square Error (NRMSE). A model with lower NRMSE value is considered to be the best predictive and has better goodness-of-fit.

## 1.4 Project Contribution

The first objective of the thesis was construction of a generic infectious disease prediction service framework that enables integration of new mathematical models. The thesis has created a generic architectural design of prediction service framework and demonstrated the possibility of integrating new mathematical models using two mathematical models.

The framework is implemented using platform independent and open source technologies, which makes components of the framework adaptable for other projects.

The thesis findings support the relevance of layered architecture combined with SOA for extensible and scalable systems, when changes are anticipated in the system requirements.

As no model will ever be completely accurate, assessing performance of the models is important. Comparing prediction results against the same reference data can provide insights into the accuracy of a certain model for a given disease. Thus, NRMSE is implemented in the prediction service to assess performance of the models.

Visualization facilitates access to prediction results and support decision making. We designed map-based interfaces to visualize the spatio-temporal prediction results. The system also visualizes user geospatial data files.

In this thesis, the spatio-temporal progression of Influenza A in north Norway is modeled as a seasonally modulated SIR(S) model.

STEM had a plug-in for the map of Norway at counties level. In order to model Influenza A at municipality level, we have created a new STEM plug-ins of Norway map with municipality level resolution. As STEM is an open source project, the new map can be reused by other researchers using STEM.

## 1.5 Organization of the Thesis

The overall structure of the report takes a form of nine chapters, including this introductory chapter.

Chapter 2: Theoretical Framework

This chapter reviews important literature that lay a foundation to our research including mathematical models, infectious disease prediction and detection systems. A brief description of the Snow system architecture and visualization is also included.

Chapter 3: Materials and Methods

This chapter describes the materials and methods used in the thesis, such as data source, software development, compartmental modeling and models evaluation.

Chapter 4: Requirements Specifications

This chapter provides rationale for the requirement specification and comprehensive description of the requirement specification.

Chapter 5: Design

This chapter describes the architectural and system design of an infectious disease prediction service based on the requirements specified in Chapter 4.

Chapter 6: Implementation and Testing

This chapter describes the implementation and testing details of the prediction service designed in Chapter 5.

Chapter 7: Mathematical Models and Evaluation

This chapter describes the method and results of SIR(S) Influenza A model and evaluation of the Bayesian and SIR(S) models.

Chapter 8: Results and Discussion

This chapter presents major findings of the thesis and discussion of the findings.

Chapter 9: Conclusion and Future Work

This chapter concludes the thesis and presents possible future work on the topic.

# Chapter 2

# Theoretical Framework

## 2.1 Introduction

This chapter reviews literature that lay a foundation for our research. The chapter begins with definition of important terms, followed by a review of mathematical models and comparison methods. Infectious disease prediction and detection systems are reviewed. A brief description of the Snow system architecture and its surveillance data is presented. A brief discussion on the impact of visualization and current health data visualization techniques are presented. Finally, the chapter concludes with a summary.

## 2.2 Terminology

Throughout this thesis the following definitions are used unless explicitly stated otherwise.

**Infectious disease informatics (IDI):** it is defined as "a sub-field of biomedical informatics concerned with the development of methodologies and technologies needed for collecting, sharing, reporting, analyzing, and visualizing infectious disease data and providing data-driven decision support for infectious disease prevention, detection, mitigation, and management" (Zeng et al. 2011).

**Surveillance:** it is defined as "systematic ongoing collection, collation and analysis of

data for public health purposes and the timely dissemination of public health information for assessment and public health response as necessary" (WHO 2005).

**Prediction:** it has two components such as forecasting and projections. A forecast is "a quantitative attempt to predict what will happen". A projection is "an attempt to describe what would happen, given certain hypotheses" (Massad et al. 2005). In this paper the term prediction is used in the sense of forecast.

**Outbreak:** it is defined as "the occurrence of disease cases in excess of what would normally be expected in a defined community, geographical area or season." It may occur in a restricted geographical area, or may extend over several countries. It may last for a few days or weeks, or for several years. A single case of a communicable disease long absent from a population or caused by an agent (e.g. bacterium or virus) not previously recognized in that area or the emergence of a previously unknown disease can be an outbreak (WHO 2012).

**Epidemic:** it is defined "an outbreak of a disease that spreads more quickly and more extensively among a group of people than would normally be expected" (Green et al. 2002).

**Pandemic:** it is defined as "an epidemic occurring worldwide or over a very wide area, crossing boundaries of several countries, and usually affecting a large number of people" (Ching et al. 2007).

**Susceptible, S:** they are Individuals susceptible to infection; they can catch the disease if they are exposed to it.

**Exposed, E:** They are infected individuals in the latent period of the disease, but not yet infectious and hence not yet able to pass the disease to others.

**Infectious (or infective), I:** they are Individuals that are infectious and capable of transmitting the infection to any susceptible they come in contact with.

**Recovered (or removed), R:** they are Individuals that were previously infected but now are neither infected nor susceptible; they have acquired immunity to infection permanently or temporarily.

## 2.3 Infectious Diseases

To clarify the discussion about mathematical models, in this section we present infectious diseases timeline that most diseases develop. As shown in Figure 2.1, usually infection timeline is divided into a series of stages starting from susceptible state. A transmission occurs when there is a contact between a susceptible and an infective, of course according to the transmission mode for the specific agent.

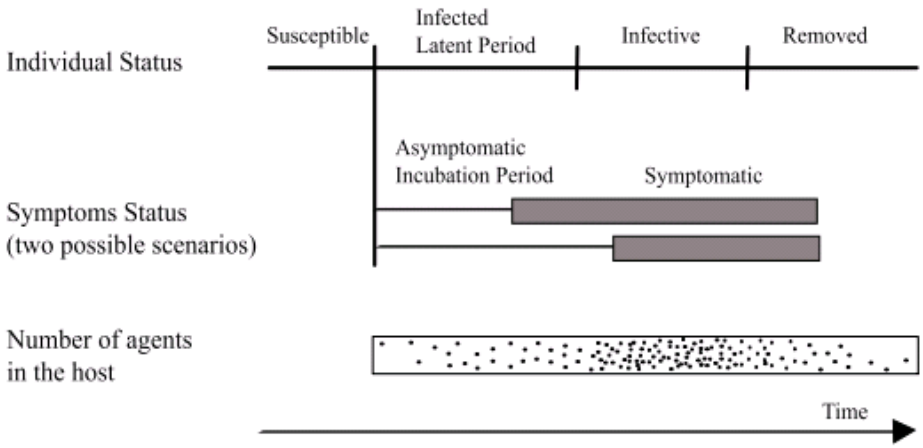



Figure 2.1: Infection-disease evolution, Source: Figure 2.1 in (Ramirez 2008)

After the host becomes infected, the agent replicates inside, so that the host becomes able to transmit the infection to others. The period between being infected and being infectious is known as the latent period, which can be as short as minutes to as long as many years for different diseases, for example Influenza has a latent period 1-4 days (White & Fenner 1994).

The period before an infected individual develop disease related symptoms is referred to as incubation period. The incubation period can be shorter or longer than the latent period, as the host can become infective before or after having any symptoms.

Acquired immunity or death caused by the infection transfer the host into removal (recovered) stage. For most viral diseases, such as measles, rubella and chicken pox the immunity can be permanent, while diseases such as Influenza the hosts could become susceptible to new strains. For diseases such as common cold and bacterial diseases the removal state does not exist since the individual re-enters the susceptible stage after recovery (White & Fenner 1994).

## 2.4 Mathematical Models

A mathematical model is an explicit mathematical description of simplified dynamics of a system (Coiera 2003). It has become an invaluable epidemiological tool in understanding the fundamental mechanism that drives the spread of infectious diseases and suggesting strategies for their control (Meyers 2007) (Grassly & Fraser 2008) (Sattenspiel 1990).

Modeling of infectious disease involves taking a set of assumptions and knowledge that defines the dynamics of disease spread. These include properties specific to a disease organism (i.e. incubation, transmission, and mortality rate) and vectors that regulate the spread of the disease (i.e. the motion of people, and waterways) (Kaufman et al. 2008). Some models predict the spatio-temporal progression of infectious diseases while others predict local temporal development of diseases (Hufnagel et al. 2004).

In epidemiology there are a number of approaches to epidemiological modeling. Currently, the three main approaches are compartmental (see section 2.4.2), agent based and contact network based (Connell et al. 2009). Contact network model captures the patterns of interactions that can lead to the transmission of infectious disease using modern methods of network theory.

In the agent-based approach the entire population and every place in the region, where people interact, is modeled as a system of software agents interacting in time and space according to prescribed rules that acts as inhabits in the city or the whole country.

This section contains a brief history of mathematical models in epidemiology and a description of compartmental and Bayesian models, which are used in the thesis.

### 2.4.1 History of Mathematical Epidemiological Models

Mathematical methods have been used in studies of communicable disease dynamics since a long time, in this section a very brief history is presented (see also (Ramirez 2008)). Daniel Bernoulli (1760) made one of the first great mathematical contributions to infectious disease control using empirical methods to examine the effectiveness of the techniques of variolation[1] against smallpox (Meyers 2007) (Gani 1980). After this,

[1]Variolation: is the old practice of vaccinating someone with the virus of smallpox to produce immunity to the disease.

late nineteenth and early twentieth century was a period important for the foundations of mathematical epidemiology (Brauer 2009).

William Farr (1852) did statistical calculations on deaths due to cholera in London (Meyers 2007) (McBryde 2006). Concurrently, John Snow (1854) made mathematical analysis to prove that the cholera was water-borne opposite to what Farr concluded. Snow's work appeared one of the founding moments of epidemiology and the use of mathematics to understand infectious diseases (Meyers 2007) (McBryde 2006).

In 1906, Hamer introduced the mass-action (homogeneous mixing) principle that has been used in chemistry. In 1927, Kermack and McKendrick formalized the principle in a deterministic model of disease transmission (Meyers 2007) (Spencer 2008).

Later, Reed and Frost (Meyers 2007) (Spencer 2008) introduced the first stochastic version of Kermack and McKendrick's model, the chain-binomial. More recently, Anderson and May (Meyers 2007) among others have extended these efforts into a flexible approach, known as compartmental modeling, for predicting the transmission of a wide range of diseases on multiple scales.

Last few decades have witnessed a tremendous progress in mathematical modeling, for example a study (Bailey 1975) referenced in (Sattenspiel 1990) has documented 539 articles on mathematical epidemiology written between 1900 and 1973. Of these papers, 336 (62%) were published between 1964 and 1973. Extrapolating this curve gives an idea of the quantity of papers found today. Another study also reported similar increasing trend between 1991 and 2005 (Keeling & Rohani 2008). A Chinese literature review (Han et al. 2009) of infectious diseases mathematical model between the period 1994 and 2006, has reported four to fivefold annual increases after 2003.

### 2.4.2 Compartmental Models

Compartmental models are commonly used models to describe the dynamics of different systems in diverse fields including epidemiology (Kaufman 2011). In epidemiology, the models subdivide host populations into different states often called compartments (i.e. susceptible, exposed, infectious, and recovered) according to their status with respect to the disease (Meyers 2007) (Kaufman et al. 2008) (Edlund et al. 2011b).

SIR, SEIR and SI models are the most common compartmental models. In these models

a set of ordinary differential equations corresponding to each compartment describes the rate of change in the size of individuals in the respective class as a result of all processes affecting this rate (Keeling & Rohani 2008).

For example let us consider the simplest version of a SIR model (Keeling & Rohani 2008) representing the passage of individuals between Susceptible (S), Infectious (I), and Recover (R) states as shown in Figure 2.2.

Let us assume homogeneous mixing and a fixed population size, $N = S + I + R$

Where $S(t)$, $I(t)$, and $R(t)$ are the numbers of susceptible, infectious, and removed in the population at time t.

The ordinary differential equation is:

$$\begin{aligned} \frac{dS(t)}{dt} &= cpS(t)I(t) \\ \frac{dI(t)}{dt} &= I(t)(cpS(t) - \gamma) \\ \frac{dR(t)}{dt} &= \gamma I \end{aligned} \tag{2.1}$$

Each contact between a susceptible and an infectious patient has a probability p of leading to transmission and contact occurs at a rate c per day. The parameter $\gamma$ is recovery rate; its reciprocal $\frac{1}{\gamma}$ determines the mean duration of the infectious period.

For a fully susceptible host population in the beginning, the initial condition for the model is $S(0) = N - e$, $I(0) = e$, and $R(0) = 0$ for a small positive e.

$$\beta = cp \tag{2.2}$$


Figure 2.2: A simple SIR model

Let us see, what factors determine whether an epidemic will occur or fail to invade? To answer this, the differential equation for $I(t)$ can be written as

$$\frac{dI(t)}{dt} = I(t)\left(\beta S(t) - \gamma\right) \tag{2.3}$$

If the initial fraction of susceptible $S(0) > \frac{\gamma}{\beta}$, then $\frac{dI(t)}{dt} > 0$ and the infection spread. This is referred to as the "threshold phenomenon". Alternatively, it can be interpreted as the result requires $\frac{\gamma}{\beta}$, the relative removal rate, to be small enough to permit the disease to spread. The inverse of the relative removal rate is called the basic reproductive number $R_0$ and is one of the most important quantities in epidemiology. It is defined as the mean number of secondary infection produced when one infected individual is introduced into a host population where everyone is susceptible.

$$R_0 = \frac{\beta}{\gamma} \tag{2.4}$$

A pathogen can spread only if $R_0 > 1$. In other words any infection that, on average, cannot successfully transmit to more than one new host does not spread.

The SIR model discussed earlier assumes that the disease spread is not affected by population births and deaths. For example, if exploring the longer-term dynamics of an infectious disease and temporary immunity are the model interest, then clearly demographic processes and immunity loss rate are also important. The SIR model can be extended to SIR(S) model (see Figure 2.3) where the host eventually returns to S state as immunity is lost.

$$\frac{S(t)}{dt} = -\beta(\frac{S(t)}{P})I(t) + \alpha R(t) + \mu(P - S(t))$$

$$\frac{dI(t)}{dt} = \beta(\frac{S(t)}{P})I(t) - \gamma I(t) - \mu I(t)$$

$$\frac{dR(t)}{dt} = \gamma I(t) - \alpha R(t) - \mu R(t)$$

Figure 2.3: SIR(S) model, Source: (Edlund et al. 2011a)

Where: $\mu$ = mortality rate

$\alpha$ = immunity loss rate

$P$ = population size

In this thesis compartmental model that come with STEM software package, which is basically based on Anderson and May's work (Kaufman 2011), is used.

### 2.4.3 Bayesian Models

Latent Gaussian models, also called Bayesian hierarchical models, are a common construct in statistical applications such as spatial and spatio-temporal models (Martino & Rue 2011) for infectious disease detection and prediction (Schrödle & Held 2011).

Markov Chain Monte Carlo (MCMC) algorithms are the standard for implementing Bayesian inference in latent Gaussian models (Martino & Rue 2011) (Manitz 2010). But, the implementation has a wide range of problems in terms of convergence and computational time. Moreover, implementation might be difficult (Martino & Rue 2011). Recently, a new approach called Integrated Nested Laplace Approximations (INLA) happens to be a promising alternative to MCMC for implementing Bayesian

inferences (Rue et al. 2009). INLA returns accurate parameter estimates in short computational time (Rue et al. 2009) (Holand et al. 2011).

We performed literature searches in *Google scholar* and *PubMed*[2] for infectious disease prediction and detection models that are implemented using INLA. We also identified additional articles from the bibliographies of included articles.

The search criteria were developed using the following key words: communicable disease, infectious disease, influenza, INLA and Bayesian model. Perhaps due to the fact that INLA is a recently growing approach, our literature search returned five results.

Manitz (Manitz 2010) developed a Bayesian model for infectious disease outbreak detection based on an algorithm in (Heisterkamp et al. 2006) and implemented using INLA, which was then applied to Campylobacter data.

Holand et al. (Holand et al. 2011) demonstrated that the INLA methodology can be used for many versions of Bayesian animal models. They also compare the inference results of INLA with MCMC. Schrödle and Held (Schrödle & Held 2011) implemented many Bayesian models describing spatio-temporal variation of disease risk using INLA. Then, they compare the models by using counts of Salmonellosis in cattle from Switzerland. Schrödle et al. (Schrödle et al. 2011) did a similar study with reported cases of bovine viral diarrhea (BVD) in cows from Switzerland, while Willgert et al. (Willgert et al. 2011) did one for Bluetongue (BT).

Geilhufe et al. (Geilhufe et al. 2012) developed a Bayesian model for spatio-temporal prediction of infectious diseases spread and implemented it using INLA. This model is used in this thesis for the prediction service constructed.

The model is primarily developed for spatio-temporal prediction of infectious diseases spread between municipalities of northern Norway which is characterized by sparse population. Physical borders and flight routes between municipalities are considered as medium for spatial spread. The model needs to be refitted recurrently before computing the next prediction by including all the previous times in the series.

The model approximates the number of disease cases $Y_{i,j}$ at a time period $i$ in munici-

[2]PubMed is a service of the U.S. National Library of Medicine that includes citations from MEDLINE and other life science journals for biomedical articles.

pality $j$ as

$$Y_{i,j} \sim Poisson(E_{i,j}\lambda_{i,j}) \tag{2.5}$$

$$\begin{aligned} Where: E_{i,j} &= \text{Offset} = \text{population in municipality j for the year of period i} \\ \lambda_{i,j} &= \text{Risk associated with municipality j in time period i} \end{aligned}$$

The log of the risk is decomposed into four components: temporal $f(t_i)$, spatial $g(s_j)$, intercept $\mu$ and unstructured component $\varepsilon_{i,j}$. Thus

$$\eta_{i,j} = \log(\lambda_{i,j}) = \mu + f(t_i) + g(s_j) + \epsilon_{i,j} \tag{2.6}$$

The temporal component $f(t_i)$ is modeled as a random walk of first order:

$$\Delta t_i = t_i - t_{i+1} \sim N(0, \tau_t^{-1}) \quad \tau_t \sim Gamma(a_t, b_t) \tag{2.7}$$

The spatial component $g(s_j)$ follows a Besag model (Besag 1974), i.e. $g(s_j)$ has the structure of a Gaussian Markov Random Field (Rue & Held 2005).

$$s_j \sim N\left(\sum_{k\varepsilon n(j)} \frac{s_k}{|n(j)|}, Q^{-1}|n(j)|^{-1}\right), \; Q = \Sigma_s^{-1} = \tau_s C, \; \tau_s \sim Gamma(a_s, b_s), \tag{2.8}$$

where:
$n(j)$ is neighboring municipalities of municipality j (i.e. sharing physical border and connected by flight routes).
C is a structure matrix of dimension $\max(j) \times \max(j)$. If two municipalities j and k are neighbors, then $c_{j,k} = 1$ and else $c_{j,k} = 0$.

The unstructured component $\varepsilon_{i,j}$ is modeled as:

$$\varepsilon_{i,j} \sim N(0, \tau_\varepsilon I), \quad \tau_\varepsilon \sim Gamma(a_\varepsilon, b_\varepsilon) \tag{2.9}$$

Equations 2.5 and 2.6 indicate that the model belongs to the class of latent Gaussian models. The predictive posterior probability distribution can be calculated as:

$$\begin{aligned} f(y_{N+1,j}|y_{1,.}, \ldots, y_{N,.}) &= \int f(y_{N+1,j}, \lambda_{N+1,j}|y_{1,.}, \ldots, y_{N,.}) \, \mathrm{d}\, \lambda_{N+1,j} \\ &= \int L(y_{N+1,j}|\lambda_{N+1,j}) f(\lambda_{N+1,j}|y_{1,.}, \ldots, y_{N,.}) \, \mathrm{d}\, \lambda_{N+1,j} \end{aligned} \tag{2.10}$$

The mean of the samples from the posterior distribution is the prediction for the next period $i + 1$ in municipality $j$.

### 2.4.4 Mathematical Models Comparison

Comparison of different candidate models is required for a range of situations including model selection and testing prediction performance of a model. Model selection is an important part of any analysis and, indeed, is central to the pursuit of science in general. Many studies have examined the question of model selection and have suggested tools for selecting the best model (Kadane & Lazar 2004).

Information-theory is an approach that attempts to identify the (likely) best model. Akaike information criterion (AIC), Bayesian information criterion (BIC) and Deviance information criterion (DIC) are widely used information criteria for selecting between competing models (Acquah & Carlo 2010).

These approaches combine a measure of the goodness-of-fit of the model to the reference data with a penalty that is a function of the complexity of the model. The complexity is proportional to the number of parameters in the model. For a given dataset, if one model fits better over another and the difference in the fit is greater than the difference in complexity, the better fitting model is preferred.

A better fit of one model over another to a given data set is a reason to prefer that

better fitting model only if the difference in fit is greater than the difference in the complexity values (Schunn & Wallach 2005).

There are various numerical measures of goodness-of-fit (Edlund et al. 2011b) (Earnest et al. 2012), such as Root Mean Square Error (RMSE), Normalized Root Mean Square Error (NRMSE), Root Mean Square Percentage Error (RMSPE), and Mean Absolute Percentage Error (MAPE).

## 2.5 Infectious Disease Prediction and Detection Systems

We performed literature searches in *Google*, articles reference lists and online databases (i.e. *PubMed*, *ACM* [3], and *IEEE Xplore*[4]). Finally, we also identified additional articles from the bibliographies of included articles.

The search criteria were developed using the following key words: communicable disease, infectious disease, influenza, bioterrorism, outbreak, simulation, prediction, detection, and software.

### 2.5.1 Infectious Disease Detection Systems

From the above search result, we presented a very brief review on the design of four outbreak detection systems that support multiple mathematical models, CASE (Cakici 2010), RODS (Tsui et al. 2003), ESSENCE II (Lombardo et al. 2003) and AEGIS (Reis et al. 2007).

For detection, RODS and ESSENCE II uses two algorithms, while CASE and AEGIS uses five and three algorithms respectively. The calculation of expected values and detection stages are combined in the descriptions of RODS, ESSENCE and CASE, while in AEGIS they are explicitly separated. Although all the systems contain multiple mathematical methods, scalability was not their design consideration except in

[3] *ACM Digital Library* is a collection of citations and full text from ACM journal and newsletter articles and conference proceedings.

[4] IEEE Xplore is a digital library providing full text access to technical literature in electrical engineering, computer science, and electronics.

AEGIS.

Regarding user interface, while all functions are integrated into one screen in AEGIS and CASE, RODS shows geographic and temporal information on different screens (EPIPLOT and MAPPLOT) and ESSENCE II also has different screens for these functions. RODS, CASE and AEGIS has email alerting capabilities, with access to a set of graphs describing the alert.

### 2.5.2 Infectious Disease Prediction Tools

From the above search result, we presented a very brief review on the design of the only one prediction system SIMID (Villa et al. 2011) and seven simulation tools, STEM (Ford et al. 2006), GLEaMviz (Broeck et al. 2011), EpiSimS (Mniszewski et al. 2008), CommunityFlu (CDC 2011*b*), FluTE (Chao et al. 2010), EpiFast (Bisset et al. 2009) and InfluSim (Eichner et al. 2007).

All the tools are used to simulate infectious diseases and also allow assessment of disease prevention, intervention, and response strategies. SIMID (Villa et al. 2011) is a web based tool for simulation of influenza, and to visualize the results of the simulation over time and space using a map-based interface. SIMID is yet available to user.

EpiSimS (Mniszewski et al. 2008), CommunityFlu (CDC 2011*b*) and FluTE (Chao et al. 2010) are agent bases infectious disease simulation tools. EpiSimS used to model multiple diseases, while the others are influenza models. In addition, in CommunityFlu the population is limited to approximately 1,000 households with 2,500 persons.

EpiFast (Bisset et al. 2009) is a contact network based tool that simulates the spread of infectious diseases across a large population.

GLEaMviz (Broeck et al. 2011) and InfluSim (Eichner et al. 2007) are compartmental model based infectious disease tools. InfluSim is an extended deterministic SEIR compartmental model based tool that simulates an influenza epidemic in a single population. GLEaMviz is a client-server simulator that allows the user to design arbitrary disease compartmental models on the worldwide scale.

STEM (Ford et al. 2006) is a standalone application that is based on an extensible software platform, which promotes the contribution of data and models by users (see section 2.5.3).

### 2.5.3 Spatiotemporal Epidemiological Modeler

Spatiotemporal Epidemiological Modeler (STEM) is an open source framework designed to provide a common modeling platform to integrate any spatio-temporal model, real data and visualization techniques to perform simulations of emerging infectious diseases (Ford et al. 2006).

The framework is written in Java programming language and runs on most platforms that support Java. STEM was designed using a component software architecture based on Equinox, which is the Eclipse implementation of OSGI standard. All of its main components (core representational framework, graphical user interface, simulation engine, disease model computations, and various data sets) are partitioned into separate bundles or plug-ins. This makes STEM extensible, flexible and re-usable (Edlund et al. 2010).

STEM was originally developed by IBM researches and contributed to the Eclipse Foundation. STEM started in the Eclipse Open Healthcare Framework (OHF) (IBM 2007) and recently promoted as a top level Eclipse Technology project (Kellen 2005).

Main features of STEM version 1.3 that are relevant to this thesis are discussed below.

#### Mathematical Models

STEM comes with SI, SIR, and SEIR compartmental models at the level of Anderson and May (see section 2.4.2). A solver is required to integrate the differential equations to determine the different states. Current release of STEM has different solvers including Finite Difference, Runge Kutta and Dormand Prince.

STEM allows modeling the spatio-temporal progress of a disease 2.4.2, but the compartmental models discussed so far deals with the trajectory of a disease in time. If we consider two locations j and k, at location j the change in the infectious population has three terms (see Equation 2.11), on site infection, infection from visitors originating at k and infections from susceptible individuals visiting site k.

$$\triangle I_j \propto \frac{\beta}{P_j} S_j I_j + \sum_k \frac{\beta}{P_j} \frac{m_{jk} P_j}{P_j + P_k} S_j I_k + \frac{\beta}{P_k} \frac{m_{jk} P_k}{P_j + P_k} S_j I_k \tag{2.11}$$

Where: $k \neq j$

$P_j, P_k$ = population at location j and k

$\beta = \beta(t)$ = the same at sites j and k

It is assumed that $m_{j,k} = m_{k,j}$ and $m_{j,j} = 1$, $m_{j,k}$ represents the set of connections between adjacent locations, air travel connections between distant sites and others.

**Denominator Data**

STEM uses graph as a framework to represent properties that define a disease spread dynamics. The nodes in a graph describe any physical location and edges describe relationships between them. Both edges and nodes can contain multiple static or dynamic labels. This graphical representation is implemented using the Eclipse Modeling Framework (EMF). Modeling such interactions as a graph allows models of infectious disease to be composed using layers of interchangeable and reusable parts (Kaufman et al. 2008).

The software comes with plugins that contain a large number of global data, for example population data, relationships between regions including nearest-neighbor and transportation systems (e.g. roads, air travel) for the 244 countries and dependent areas defined by International Standards Organizations. Of these, county level geographic data, air transportation model and population data (2006 census) of Norway are included.

**Transportation Models**

In current release of STEM the mixing rate between adjacent locations is scaled by a characteristic mixing distance, $\delta_0$, for each region. This parameter models the distance people travel, on average, in a day. In this model the fraction of people leaving site j for any neighboring site k is shown in Equation 2.12.

$$m_{jk} \sim \min\left(1.0, \frac{\delta_0}{\sqrt{A_j}}\right) \tag{2.12}$$

Where: $\delta_0$ is determined by model fitting.

$A_j$ is area of location $j$

Air transport is modeled as a simplified "pipe" model (Lessler et al. 2009), in which individuals flow in and out of the air transport system is based on the number of arrivals and departures from a given airport, different from a fully saturated model where all routes are modeled individually. However, the architecture STEM' also supports the creation of alternative air travel plugin.

**External Data Source**

STEM allows importing external files in csv format for analysis or playback. For import, STEM expects a csv file corresponding to each disease state in the model; and determines the type of model by checking which files are available. Even, on the occasions where the data on every state is not available (i.e. data from public health surveillance) the files and file headers must be included. For example, if STEM finds a csv file of S and I states it will assume the model is of type SI.

**Visualization**

STEM visualizes simulations of the geographical spread of infectious diseases using its internal map view and overlay it on Google Earth. The internal map view is a light weight build that allows a user to watch a simulation at run time. The Google Earth interface enables the logging of the simulation data (in the form of KML files) and displayed on Google Earth by mapping disease states to color intensity (Kaufman et al. 2009).

**Automatic Experiment perspective**

STEM enables to determine the parameter values for a good fit of the output of a disease model to a reference data. An automatic experiment run a sequence of simulations, varies parameters of the model for each simulation and compares the simulation results with the reference data.

STEM has a pre-configured optimization algorithm called Nelder-Mead simplex (Nelder & Mead 1965). The algorithm searches model parameter space to minimize a functional error measurement, normalized root mean square error (NRMSE) between the incidence, $\widehat{I}$, predicted by the simulation and the historic reference data, I.

NRMSE is calculated (see equation 2.13) as root mean squared error over all prediction time periods and normalized by the difference between the maximum and minimum

aggregated reference data number of incidence.

$$NRMSE(\widehat{I}, I) = (\max \theta_i - \min \theta_i)^{-1} \sqrt{T^{-1} \sum_i \left(\widehat{\theta_i} - \theta_i\right)^2} \tag{2.13}$$

Where: $I$ = reference incidence count

$\widehat{I}$ = Predicted incidence count

$i$ = prediction time period

$j$ = set of all locations common to both the simulation and reference data.

$\theta_i = \sum_j I_{i,j}$ = aggregated reference incidences count at period i

$\widehat{\theta_i} = \sum_j \widehat{I}_{i,j}$ = aggregated predicted incidences count at period i

$T$ = set of all periods for which there is reference data

**Analysis perspective**

Analysis perspective is one of the five perspectives in STEM. It contains tools that help users to perform analysis, fitting, model comparison, and validation across multiple simulations and data sets. The tools can be used either with a complete simulation or external data source (i.e. bio-surveillance) in csv format.

STEM currently enables the following types of analysis tools:

1. **Estimator:** Estimate parameters for a (new) model from an existing data set. Algorithms for parameter estimation using SI, SIR, and SEIR models are included in the current release.

2. **Epidemic:** Aggregate data across locations for a given epidemic scenario and plots the aggregated data.

3. **Epidemic:** Aggregate data across locations for a given epidemic scenario and plots the aggregated data.

4. **Scenario Comparison:** Measures the RMS difference between two existing data sets.

5. **Lyapunov Analysis:** Compare two existing scenarios or data sets based on their trajectories in a Lyapunov Phase Space.

#### STEM Headless

STEM enables to run simulations from command line simply by supplying the -headless command line argument. For example, to run a simulation and log the simulation output in csv format.

*stem -headless -log /var/log -uri platform:/resource/Norway/scenarios/ NorwayInfluenzaAScenarioLevel2.scenario*

#### Logger

The current distribution of STEM contains csv and map loggers that provide users with the ability to output disease state data to a csv and map files. STEM Loggers have the ability to select specific disease/population model compartments to record.

## 2.6 The Snow Agent System

#### Background

In Norway, all healthcare service providers (i.e. GPs and hospitals) are connected via a national health net, an independent and secure IP-network, which enables electronic communication between participating institutions (Hartvigsen & Pedersen 2012).

#### Architecture

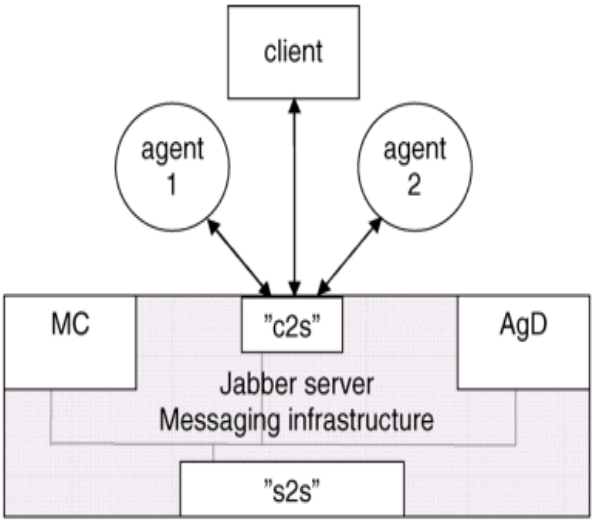
The Snow Agent system is a distributed disease surveillance system that extract and correlate data from multiple electronic health record (EHR) systems (i.e. hospitals, GP offices, and laboratories), in which queries are run against distributed, in-situ data (Bellika et al. 2009). This approach is classified as a third-generation data-integration system (Lober et al. 2004).

The Snow Agent System consists of an overlay network of Snow Agent System (SAS) servers, which is an extension to Jabber extensible open source server that implements XMPP. In the XMPP based routing overlay network each participating institutions connects to a SAS server which has access to the EHR system. SAS servers within a region, like a county, connect to a common Post Office (PO) server which is in the

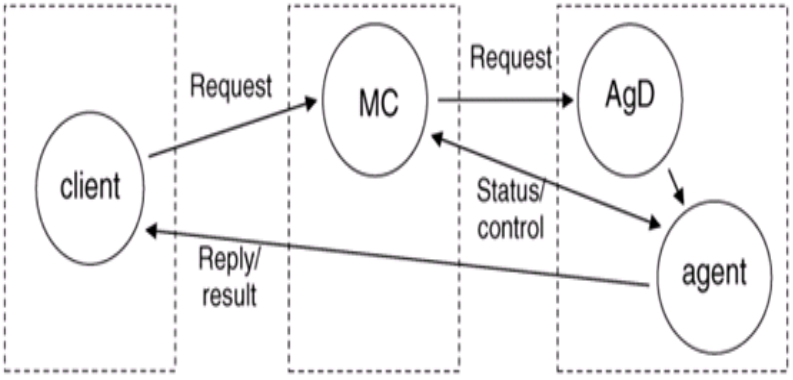
health net. The PO servers of different regions connect to one another in order to facilitate global message delivery (Bellika et al. 2007).

A SAS server contains a mission controller (MC) and agent daemon (AgD) components as shown in Figure 2.4(a). MC performs mission control by receiving mission specification (XML message) from client and negotiates it with remote AgDs.

AgD instantiate processes named mission Agent that performs a series of operations based on a specification received from a MC. A mission agent may employ sub-missions by sending a mission specification to a MC. Finally, the agent sends mission results directly to the mission requester using ordinary XMPP messages.




(a) Main components




(b) Interactions between components

Figure 2.4: The Snow Agent System, Source: (Bellika et al. 2007)

### Infectious Disease Surveillance[5]

The Snow agent system contains one participating laboratory, microbiology laboratory at University Hospital of Norway (UNN), which is the regional hospital of the northern health region. The surveillance covers various diseases of respiratory and gastrointestinal disease groups covering municipalities of Troms and Finnmark counties of the region.

For the data collection, as shown in Figure 2.5, an XMPP client running on the PO server sends mission specification to the MC at the PO server, which negotiates with the AgD in the SAS server at UNN. The AgD create a mission Agent and the results go back to the client. Finally, the client sends the result to a server running at UiT. Norwegian confidentiality laws prohibit centralization of patient information (Hartvigsen & Pedersen 2012); as a result all missions are limited to weekly aggregated data.

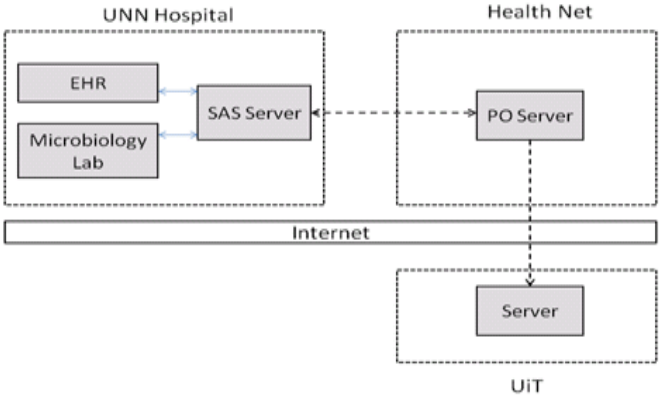



Figure 2.5: Snow Agent System laboratory data extraction from UNN

The laboratory result sent from the PO server is stored as XML file(see Figure 2.6) in reportdb MySQL database on the server at UiT. The schema of "reports" table that stores the lab data is as shown in Table 2.1.

[5]Can be accessed at: http://snow.cs.uit.no/

Table 2.1: Schema of the Snow database reports table

| Field | Type | Sample value |
|---|---|---|
| code | Varchar | 19 |
| symptom_group | Varchar | Luftvei |
| subcategory | Varchar | Forkjolelsesvirus |
| content | Mediumtext | <XML report content> |
| last_update_date | Date | Wed 14 Oct 09 |

```
...
<subAreas>
  <area type="MUNICIPALITY" code="1901" name="Harstad" population="23126">
    <onLocation type="patient">
      <DataCollection/>
      <AggregatedCollection>
        <DataSet daysPerUnit="7" startDate="2007-05-29">
          <numberOfUnits>4</numberOfUnits>
          <dataResults>
            <result>
              <name>Result=Influenza A</name>
              <values>1, 0, 3, 0</values>
            </result>
          </dataResults>
        </DataSet>
      </AggregatedCollection>
    </onLocation>
    <onLocation type="requester">
      ...
    </onLocation>
  </area>
  ...
</subAreas>
...
```

Figure 2.6: A fragment of Snow XML report

## 2.7 Visualization

The impact of health information is often less related to the accuracy of the information than to the fact that information is not routinely used by the relevant stakeholders, to some extent, due to communications gap (Evaluation 2009).

Visualization exploits our natural ability to recognize and understand visual patterns and increase the amount of useful information that decision-makers extract from com-

plex and/or voluminous data sets (Li et al. 2001) (Dull & Tegarden 1999) (Burstein et al. 2008). Accordingly, translating the specific outcome from decision support systems (DSS) into charts, maps, and other graphical displays makes data interpretation much more intuitive.

Researchers from Human-Computer Interaction (HCI) and other disciplines have been examining information visualizations effect on user satisfaction, the effort or time it takes to complete tasks aided with the technology, and decision accuracy (Kellen 2005). Various visualization techniques have been developed, but applying visualization in the context of decision-making is not well understood (Burstein et al. 2008).

A cognitive fit theory was introduced to explain the numerous equivocal results from decades of studies on information visualization using graphs and tables (Vessey & Galletta 1991) (Zhang 2006). The theory proposes that the correspondence between the decision task and information presentation format leads to superior decision performance for individual users.

## Spatiotemporal Health Data Visualizations

Studies have revealed strong spatial aspect of diseases spread; this can be traced back to 1694 Tom Koch map for plague outbreaks (Gao et al. 2008) and 1854 John Snow map (see Figure 2.7) for cholera outbreak (Frerichs 2006).

Figure 2.7: John Snow map for describing the Broad Street pump cholera outbreak of 1854 (Frerichs 2006)

Current spatio-temporal health data visualizations utilize geographic information systems (GIS) (Reinhardt et al. 2008) such as health maps implemented using Google Maps API, as shown in Figure 2.8, (Yi et al. 2008) (Freifeld et al. 2008), Google Earth KML (Kaufman et al. 2009), MSN Virtual Earth (Gao et al. 2008), arcIMS (Tsui et al. 2003) and R (Yi et al. 2008).


Figure 2.8: Screenshot of HealthMap, Source (CDC 2011*a*)

## 2.8 Summary

The chapter presented brief explanation of key terms in the thesis, and followed by a description of each state in infectious diseases timeline that most diseases develop.

An introduction to mathematical modeling in epidemiology and its historical background is briefly presented. Compartmental modeling, which is one of the main modeling approaches in epidemiology, is described with a SIR model example. Then, a short review of Bayesian models implemented using INLA and detailed description of a model, which is used in the thesis, is presented. Finally, the section ends with a brief discussion on mathematical model evaluation techniques such as error measures and Information theory.

From a small review on infectious disease prediction and detection services, a comparison of four detection systems and seven prediction tools is presented. The chapter followed by a broad description of STEM and its main functionalities. An architectural overview of the Snow system, which is the data source for this thesis, is described. The system has an architecture that can be classified under third generation data integration systems, where data queries are performed against in-situ distributed EHR systems. The chapter also discusses how the Snow system gets laboratory data of infectious diseases from the microbiology laboratory at UNN.

Finally, the chapter briefly discussed researches that address the effect of visualization on increasing decision-makers ability to extract useful information from complex and/or voluminous data sets. Current spatio-temporal health data visualizations are also briefly reviewed.

# Chapter 3

# Materials and Methods

## 3.1 Introduction

This chapter describes the materials and methods used in the thesis work. The chapter starts with a description of materials and method used to develop the software for the infectious disease prediction service framework. Methods used to construct compartmental models and a systematic comparison of the models with Bayesian model is explored. Finally, the chapter concludes with a summary.

## 3.2 Materials

### 3.2.1 Hardware and Software

The following hardware and software tools that has been used in the thesis is presented in Table 3.1.

Table 3.1: Software and hardware materials

| Software | Hardware |
|---|---|
| Eclipse Helios v 3.7.0 | Windows 7 Laptop |
| STEM v 1.3 | Windows 2008 Server |
| R v 2.13.1 and packages | |
| UMLet v 11.5 | |

### 3.2.2 Study Area and Data Source

Northern Norway consist of three counties Nordland, Troms and Finnmark; each with 44, 25, and 19 municipalities respectively. The region covers about 35% of the Norwegian mainland (Statistics Norway 2002) with a population of 468,251, which is about 9.5% of the total population of Norway (Statistics Norway 2011).

As discussed in section 2.6 the Snow system is an infectious disease surveillance system covering Troms and Finnmark counties of the region. For this thesis we have used weekly aggregated laboratory confirmed cases of Influenza A, Norovirus, RS-virus, Mycoplasma pneumoniae, and Rhinovirus.

### 3.2.3 Data Analysis

The Influenza A dataset contains cases from Dec 2007 to Apr 2012 registered in five seasons. The peak months of the Influenza A activity was in Nov, Jan and Feb. During 2009/10, the Influenza A activity peaked twice because of the 2009 H1N1 pandemic. The activity peaked once in the spring, when the 2009 H1N1 virus first emerged, and again in Nov, when the region went through its regular Influenza A season.

The Norovirus and Mycoplasma pneumoniae datasets contain cases from May 2007 to Apr 2012, while the Rhinovirus dataset contains cases from Apr 2009 to Apr 2012 and RS-virus from Dec 2007 to Apr 2012.

## 3.3 Software Development

Construction of a prototype that demonstrates the solution to the research problems was done using a method modified from an engineering approach described in (Denning et al. 1989). The method follows an iterative and incremental approach where the entire development process is composed of activities such as:

1. State requirements;
2. State specifications;
3. Design and implement the system;

4. Test the system.

The requirements analysis, specification, and design of architecture are defined using the Waterfall approach, followed by iterative and incremental implementation. The implementation was split into a series of consecutive sub tasks time boxed from five to ten days. Each sub task contained a small set of features from code to test and usually some input to the design and requirement.

### 3.3.1 User-Centred Design

Personas has been used as a user-centred design approach. Through a process of analysis and refinement, potential users of the system are represented by four fictitious characters (see section 4.4).

### 3.3.2 Requirement Specification

Volere Requirements Process and its associated Specification Template (Robertson & Robertson 1999) has been used as the basis for gathering, confirming, and documenting the requirements.

## 3.4 Mathematical Models

A detail description of the methods used in the mathematical modeling is described in Chapter 7. Here we presented a brief description of the methods.

**Compartmental Model**

We model SIR(S) Influenza A model for municipalities in the two counties of Norway using STEM with a seasonally modulated transmission coefficient and air transportation model between the municipalities.

We have used Influenza A laboratory results from Jan 2008 to Apr 2010, to fit the model. The number of cases from the laboratory represents only a fraction of the total incidence, thus we made an assumed that the reporting fraction is 3%.

The rate of immigration and emigration were estimated using data from the Norwegian statistics bureau (Statistics Norway 2011). Immigration and emigration were considered to be constant across all the municipalities.

**Bayesian Model**

We have applied the model to various infectious diseases (i.e. Influenza A, Rhinovirus, Mycoplasma Pneumoniae, RS-virus and Norovirus) for weekly and monthly (defined as four weeks) predictions for about two years and calculated the NRMSE.

**Influenza A Models Comparison**

For the comparison of the compartmental and Bayesian models (both weekly and monthly predictions) we calculate the NRMSE for each model across about two years predictions against unseen data. The model with the lowest NRMSE is considered to have the best predictive ability and represent better goodness-of-fit.

## 3.5 Critique of the Methods Used

The use of personas as a participatory design technique and compromise between the actor and on-site user is well known. However, if the analysis is not careful, confidence in the resulting personas will be undermined and the design direction can be inappropriate.

## 3.6 Summary

The chapter presented a list of hardware and main software tools used in the thesis. The thesis has used infectious diseases laboratory data collected by the Snow system, which covers Troms and Finnmark counties of northern Norway.

A discussion on the methods used for prototype development process, SIR(S) modeling and evaluation of the model with a Bayesian model are presented. Finally, the chapter concluded with a critique of the methods.

# Chapter 4

# Software Requirements Specification

## 4.1 Introduction

This chapter provides a comprehensive description of the requirements specification for an infectious disease prediction service mainly focusing on formulating a framework for integrating and comparing mathematical models. We begin by an overall description of the system including assumptions, dependencies and potential users of the system. Then it is followed by, a rationale for requirement specification method selection and source of requirements. Functional and non-functional requirements specification of the whole system is described next. Finally, the chapter concludes with a summary.

## 4.2 System Description

The infectious disease prediction service is a framework that:

- enables integration of new mathematical models.
- enables comparison of models.
- visualizes prediction results in a way that facilitates prediction results interpretation and decision making.

The prediction service uses laboratory data that are already being collected by the Snow system to provide timely infectious diseases forecasts that appear as current weather forecasting services do.

**Constraints**

The product had the following constraints:

- The service shall use weekly aggregated laboratory data from the Snow system.
- The service shall give data to other applications (e.g. decision support systems).

**Naming Convections and Definitions**

Decision Support system=DSS (i.e. Diagnosis and travel decision support systems)
General Practitioner=GP
Microsoft Solutions Framework =MSF
Persona is an "archetype of a fictional user representing a specific group of typical users (Miller & Williams 2006)."

**Users**

We assume the system could have many potential user groups with diverse benefit and requirements from the system. The main users of the system include:

- Epidemiologists
- Mathematicians/Statisticians
- Healthcare professionals (i.e. general practitioners, pharmacists, laboratory professionals)
- Public

## 4.3 Requirements Process and Specification Method

A wide variety of requirements specification techniques have evolved over the years, each with its own recognized strengths and weaknesses. But some techniques are more appropriate for a given project than others. In a project with an iterative approach, all iterations bring new requirements and changes to existing one. Thus, a method

that helps to revise the requirements, and track the changes is necessary. Due to these reasons the Volere Requirements Process and its associated Specification Template (Robertson & Robertson 1999) was chosen for this thesis as the basis for gathering, confirming, and documenting the requirements.

## 4.4 Source of Requirements

Certainly, user involvement is an important element in the success of a project (Miller & Williams 2006). As infectious disease prediction service is a recently emerging area, it was not possible to get either requirements defined for such service or dedicated users available for the duration of the project. As a result, we have used personas as a participatory design technique and as a compromise between the actor and on-site user. Four personas representing potential user of the system are documented using Microsoft Solutions Framework (MSF) Agile Persona Template (Miller & Williams 2006).

Other sources of requirements were meetings with professionals. We had three unstructured meetings with two mathematicians/statisticians and one epidemiologist separately. The mathematicians/statisticians are researchers experienced in epidemiological modeling. The epidemiologist also practised as a general practitioner.

Requirements specified in CASE (Cakici 2010), RODS (Tsui et al. 2003), ESSENCE II (Lombardo et al. 2003) and AEGIS (Reis et al. 2007) were also source of requirements. The requirement gathering process continued during design and implementation stages resulting more requirements and adjustment of previous requirements as well.

### Users Personas and Characteristics

The archetype of typical fictional users representing epidemiologist, mathematician, GP, and the public is presented below. In addition to descriptive roles, we have assigned names (name of pioneers of the corresponding role) to each of the personas, to make it easier to refer to them later in the thesis.

Table 4.1: Persona of Epidemiologist

| | |
|---|---|
| **Name:** Dr. J. Snow | **Status and Trust Level:** High |
| **Role:** Infectious Disease Epidemiologist | **Demography:** Male |
| **Knowledge, Skills, and abilities:** Dr. Snow has detailed knowledge of infectious disease dynamics and the implications of infectious disease spread on both current and future problem-solving and decision-making. He understands and uses the power of mathematical and statistical methods to get insight into how infectious diseases spread in the future and the difference in accuracy between mathematical models. | |
| **Goals, motives, and concerns:** Get to know how infectious disease spread some time ahead, conduct investigations of infectious disease spread and potential mitigation strategies, and report investigation results. | |
| **Usage Patterns:** Dr. Snow uses the system always. | |

Table 4.2: Persona of Mathematician

| | |
|---|---|
| **Name:** Daniel Bernoulli | **Status and Trust Level:** High |
| **Role:** Mathematician | **Demography:** Male |
| **Knowledge, Skills, and abilities:** Bernoulli has deep skills in mathematical modeling and knowledge of infectious diseases dynamics. He also has knowledge and experience in developing software. | |
| **Goals, motives, and concerns:** Test mathematical models, visualize prediction results and compare mathematical models. | |
| **Usage Patterns:** Bernoulli uses the system every time he is building or improving mathematical model. | |

Table 4.3: Persona of General Practitioner

| | |
|---|---|
| **Name:** Dr. Hippocrates | **Status and Trust Level:** High |
| **Role:** General Practitioner | **Demography:** Male |
| **Knowledge, Skills, and abilities:** Hippocrates has skills to diagnose, treat, and help prevent diseases. He has the ability to work under pressure and make quick and clear decisions. | |
| **Goals, motives, and concerns:** Apply new knowledge and provide high quality care. | |
| **Usage Patterns:** Hippocrates has a busy schedule and can't spend much time on accessing the system. | |

Table 4.4: Persona of Public

| | |
|---|---|
| **Name:** Jennifer | **Status and Trust Level:** Medium |
| **Role:** Public | **Demography:** Female |
| **Knowledge, Skills, and abilities:** ePatient | |
| **Goals, motives, and concerns:** Gather information about a medical condition of particular importance to her children. | |
| **Usage Patterns:** She uses the system when her children develop some symptoms. | |

## 4.5 Functional requirements

This section describes the functional requirements of the prototype. We have determined the following requirements from the requirement sources:

Table 4.5: Functional Requirement one

| | |
|---|---|
| **Requirement #:** 1 | **Event/Use case #:** 1 |
| **Description:** The product shall record data from the Snow system. | |
| **Rationale:** To make the data accessible to the system. | |
| **Source:** Requirement analyst | |
| **Fit Criterion:** The data shall be accessed by the system components. | |
| **Dependencies:** None | **Conflict:** None |
| **History:** v 1.0 | |

Table 4.6: Functional Requirement two

| | |
|---|---|
| **Requirement #:** 2 | **Event/Use case #:** 2 |
| **Description:** The product shall make periodic prediction. | |
| **Rationale:** Periodically make prediction based on expert specification | |
| **Source:** Dr. Snow, Dr. Hippocrates, and Jennifer | |
| **Fit Criterion:** The prediction result shall agree with the schedule | |
| **Dependencies:** 3 | **Conflict:** None |
| **History:** v 1.0 | |

Table 4.7: Functional Requirement three

| | |
|---|---|
| **Requirement #:** 3 | **Event/Use case #:** 3 |
| **Description:** The product shall allow configuration of prediction properties. | |
| **Rationale:** To easily update system properties. | |
| **Source:** Dr. Snow | |
| **Fit Criterion:** The change in the configuration affects the prediction. | |
| **Dependencies:** None | **Conflict:** None |
| **History:** v 1.0 | |

Table 4.8: Functional Requirement four

| | |
|---|---|
| **Requirement #:** 4 | **Event/Use case #:** 4 |
| **Description:** The product shall allow users to schedule a prediction for a particular disease. | |
| **Rationale:** Enable users to investigate a particular diseases of their choice. | |
| **Source:** Dr. Snow and Bernoulli | |
| **Fit Criterion:** The prediction result shall agree with the schedule | |
| **Dependencies:** None | **Conflict:** None |
| **History:** v 2.0 | |

Table 4.9: Functional Requirement five

| | |
|---|---|
| **Requirement #:** 5 | **Event/Use case #:** 5 |
| **Description:** The product shall allow evaluation mathematical models. | |
| **Rationale:** Enable users to evaluate the performance of mathematical models. | |
| **Source:** Dr. Snow and Bernoulli | |
| **Fit Criterion:** Users shall be able to get evaluation results. | |
| **Dependencies:** None | **Conflict:** None |
| **History:** v 1.0 | |

Table 4.10: Functional Requirement six

| | |
|---|---|
| **Requirement #:** 6 | **Event/Use case #:** 6 |
| **Description:** The product shall allow visualization of external data from users. | |
| **Rationale:** Enable users to visualize their (simulation or real) data. | |
| **Source:** Dr. Snow and Bernoulli | |
| **Fit Criterion:** Users shall be able to download the result. | |
| **Dependencies:** None | **Conflict:** None |
| **History:** v 2.0 | |

Table 4.11: Functional Requirement seven

| | |
|---|---|
| **Requirement #:** 7 | **Event/Use case #:** 7 |
| **Description:** The product shall allow other applications to access prediction results. | |
| **Rationale:** Enable decision support systems to access prediction results. | |
| **Source:** Requirement analyst | |
| **Fit Criterion:** Users shall view the result. | |
| **Dependencies:** None | **Conflict:** None |
| **History:** v 2.0 | |

## 4.6 Use Case

From the above functional requirements a UML use case diagram (see Figure 4.1) is developed to define the prototypes boundary and the actors involved in each use case. An active actor is represented as a stick figure and autonomous actors as a box.

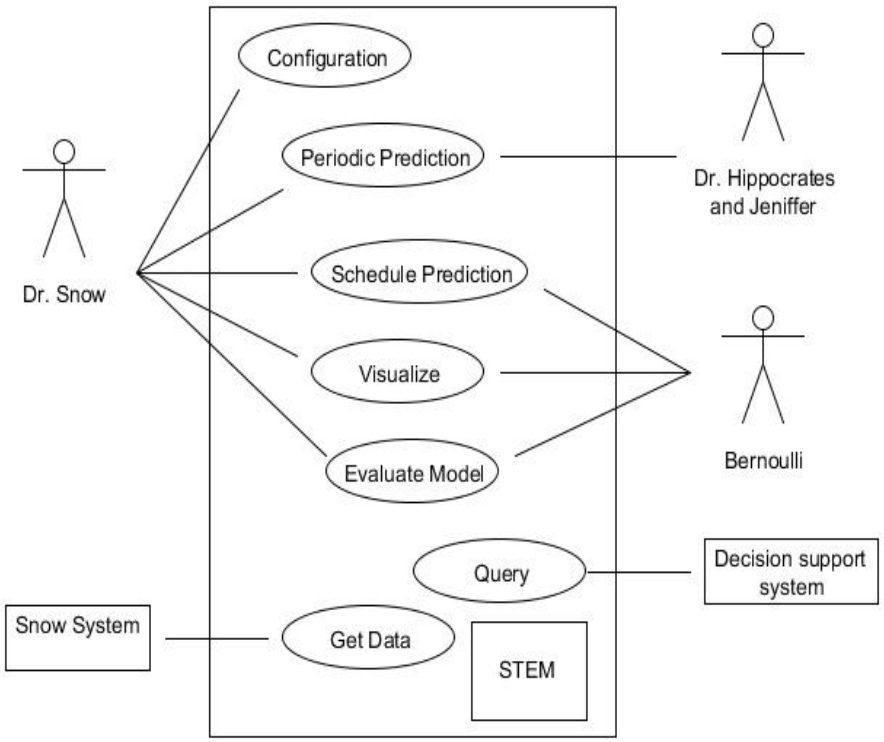



Figure 4.1: Use case diagram for the Infectious disease Prediction Service Framework

Further description of each use case in the system boundary is presented as follow:

**Use Case#:1** Get lab data
*Purpose:*
Read laboratory data from Snow system and store in the system.
*Typical flow of events:*

- System check calendar for data extraction.
- System reads the data from the Snow database.
- System formats the data.
- System stores the data.

**Use Case#:2** Periodic Predictions
*Purpose:*
Make time triggered prediction of selected infectious diseases using predefined mathematical models.

*Typical flow of events:*

- System checks prediction calendar.
- System makes prediction.
- System stores prediction result.
- System visualizes prediction results.

**Use Case#:3** Configurations
*Purpose:*
Allow users to control the system properties such as selecting diseases, mathematical models, and model parameters.
*Typical flow of events:*

- User changes the prediction service property.
- System uses the change.

**Use Case#:4** Schedule predictions
*Purpose:*
Allow users to schedule a prediction for a particular disease.
*Typical flow of events:*

- User inputs the schedule.
- System saves the prediction.
- System runs the scheduled prediction.
- System visualizes the prediction.
- User accesses the visualization.

**Use Case#:5** Evaluate Mathematical Models
*Purpose:*
Allow users to assess performances of mathematical models.
*Typical flow of events:*

- User chooses models for evaluation.
- User chooses a comparison method.

- System makes a comparison.
- System visualizes the result.

**Use Case#:6 Visualize External Data**
*Purpose:*
Allow users to visualize their data.
*Typical flow of events:*

- User upload data.
- System returns the visualization of the data.

**Use Case#:7 Query Prediction Results**
*Purpose:*
Allow other applications to access prediction results.
*Typical flow of events:*

- Applications inputs search criterion.
- System searches the database.
- System returns the result.

## 4.7 Non-functional requirements

### 4.7.1 Scalability

The system shall scale to incorporate mathematical models, comparison algorithms, and data from bio-surveillance systems.

### 4.7.2 Extensibility

The system shall enable easy extension of functionalities to meet new requirements of users as the system grows.

### 4.7.3 Usability

The system shall disseminate prediction results in a way that enable users to easily interpret and make decision.

## 4.8 Summary

We discussed why the Volere requirements process and specification method is suitable for the thesis. Personas is used as a participatory design technique, since we couldn't get dedicated users available for the duration of the project. Personas that define characteristics of typical fictional users representing potential users such as epidemiologist, mathematician, GP, and public are created using Microsoft Solutions Framework (MSF) Agile Persona Template.

Other sources of requirements were discussion with two mathematicians/ statisticians and requirements specified for infectious disease detection systems.

The chapter contains a UML use case diagram that defines the system boundary and the actors involved in each use case. Functional requirements are described and specified using the Volere requirement specification template. Finally, the chapter ends with a specification of non-functional requirements.

# Chapter 5

# Design

## 5.1 Introduction

This chapter describes the architectural and system design of an infectious disease prediction service based on the requirements specified in chapter four. As the aim of the thesis is construction of a framework, the design focused on common parts of infectious disease prediction service skeleton, and provides a range of reusable components in order to ensure extensibility and scalability.

We begin by an overall description of design considerations. Architectural design and design rational is described followed by data, component designs, and user and component interfaces designs. Finally, the chapter concludes with a summary.

## 5.2 Design Considerations

**Assumptions**

- The user of the system is aware of basic operations of a computer and web pages and the standard terms used.
- The prediction system runs on the same machine as the Snow system, which is the data source for the prediction.

- Data from the Snow system is anonymized, geocoded and clean. Consequently, no pre-processing is required.

**System environment**

The prediction service is designed to run on a Snow server at University of Tromsø running Linux operating system. It should also be possible to run on other platforms, including Windows.

## 5.3 Architectural Design

Software architecture design is the process of defining a structured solution that meets all of the functional requirements, while optimizing non-functional requirements. It involves a series of high level decisions that can have considerable impact on the overall success of the application.

The architecture of the system is designed using three layered architecture combined with service-oriented architecture (SOA) (High Jr et al. 2005), where modules on business and data layer exposes their functionality as a service.

The modular design is adaptive to change and enables to scale across new mathematical models, model comparison algorithms, and data sources. The design also enables individual modules to be upgraded with inexpensive, off-the-shelf application software.

The SOA approach supports loose coupling between modules and makes services reusable. As a result, the system supports different software clients, which is especially important as the system could give service to decision support systems.

The architectural design on Figure 5.1 shows the modules that compose each layer and interconnection between modules.

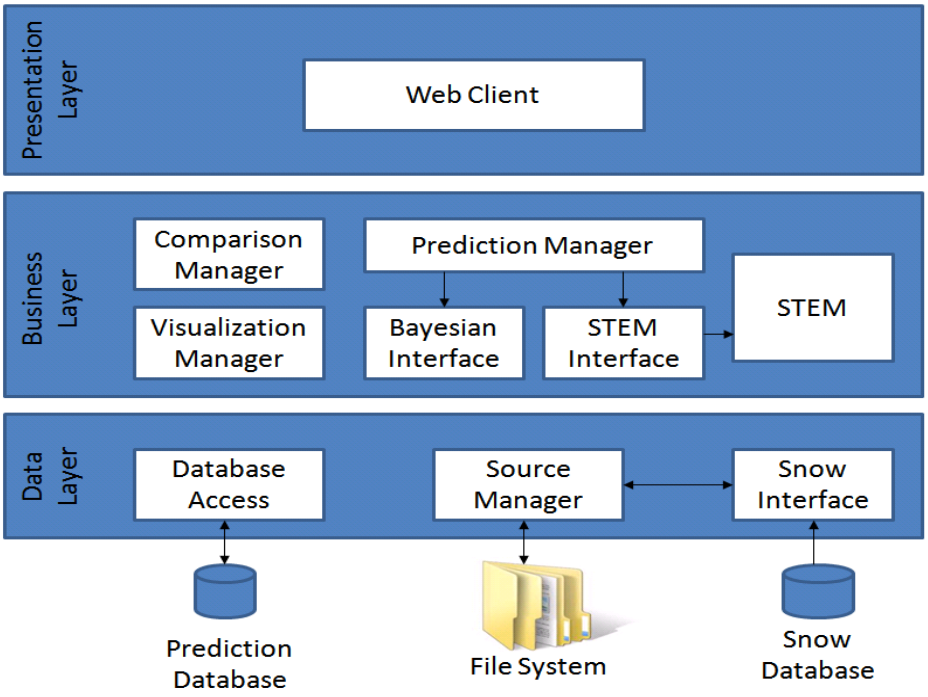



Figure 5.1: Infectious Disease Prediction Service architecture

## Decomposition Description

The system was designed using a 3-layered architecture consisting of data layer, business layer and presentation layer. The web client on the presentation layer interacts with components on the business layer through web service interfaces. Components on the business layer interact with the database and static files to perform all business operations.

The data layer consists of data store and access modules to handle data such as prediction schedule, prediction results and laboratory data. The Source Manager handles the reading and formatting of public health data, such as laboratory data from the Snow system. Data from the external source is passed through an interface module which handles the formatting of source data. Since the operation of data interpretation module is self-contained and strictly defined, the addition of new data sources is straight- forward.

Database access module provides a web service interface to access schedule information and prediction results in the database.

The business layer consists of Prediction Manager, Visualization and Comparison Manager. Prediction Manager is used to manage all the predictions in the system. The manager read prediction schedules and orders a prediction using a model interface module that correspond a particular model. Input/output data processing and model execution is contained in the model interface modules, which makes addition of new model straightforward.

Visualization module provides standard web service interfaces to generate maps for visualization of prediction results. The visualization request can be obtained from clients such as models interfaces and web client.

The Comparison Manager provides web service interfaces to compare mathematical models. The module is extensible to include different comparison techniques. In this thesis we have developed a module that implements NRMSE.

The presentation layer is an interactive web client that enables user interaction with the system, such as access to current prediction results, prediction scheduling, comparison of models and user data visualization.

## 5.4 Data Design

### Data Description

The information domain of the system consists of laboratory data, infectious disease prediction schedule and results, and maps. It also consists of public health data that is upload by the user for visualization. The system has two data stores: database and static files. The prediction schedule and results are stored in the database. The laboratory data, maps, and user data are stored as a static file.

The data stored in the database is organized in two tables, schedule and prediction. The schedule table store scheduled prediction information and the prediction table stores prediction results.

### Data Dictionary

```
CREATE TABLE Schedule
(
id int(10) NOT NULL auto_increment,
Infectous_agent Varchar(50),
date date,
model Varchar(50),
period Varchar(20),
PRIMARY KEY (id)
);

CREATE TABLE Prediction
(
prediction_Id int(10),
content Mediumtext,
PRIMARY KEY (prediction_Id)
);
```

## 5.5 Components Design

### SNOW Interface

The Snow interface module handles the extraction and formatting of source data. The Metadata class is specification of the metadata required for the data extraction and formatting.

The Manager Class implements main (), monitor (), and cleanLabData () methods for controlling the interface operations control. The Database Class implements a method that performs extraction of the lab data from the Snow database using JPA entity class. The CSVFile Class implements methods that perform basic operation on csv file.

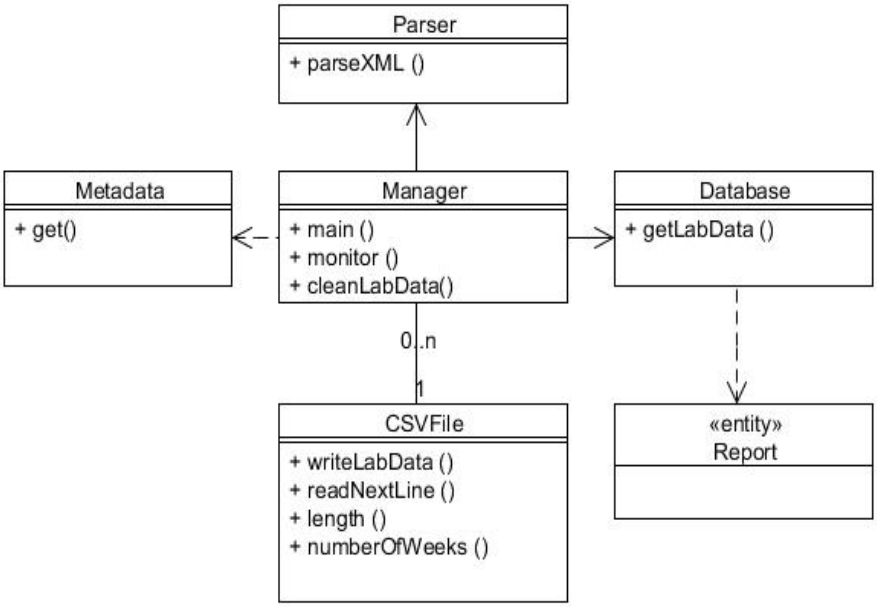



Figure 5.2: Snow Interface Class Diagram

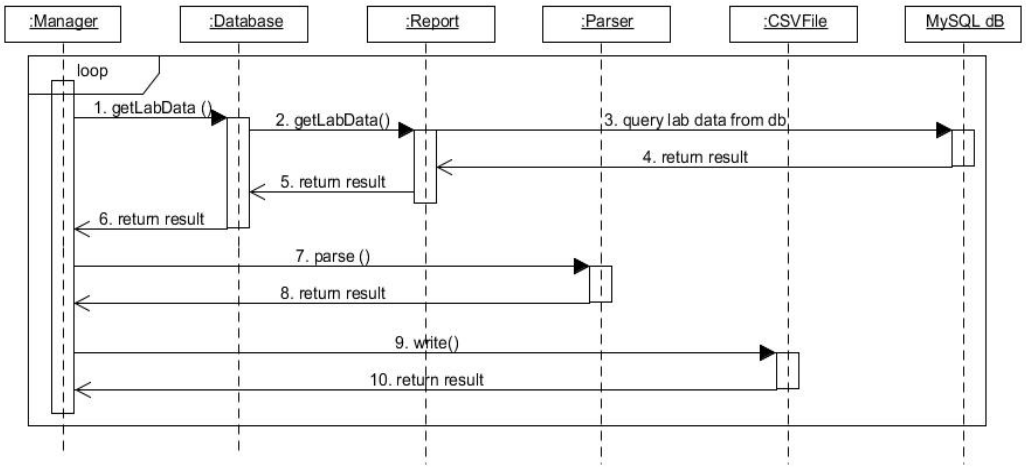



Figure 5.3: Snow Interface Sequence Diagram

### Database Access

Database access module provides a web service interface that allows access to prediction schedules and results in the prediction system database. dBService is JAX-RS resource class. The Database class implements methods that perform insert and query operations, to/from database.

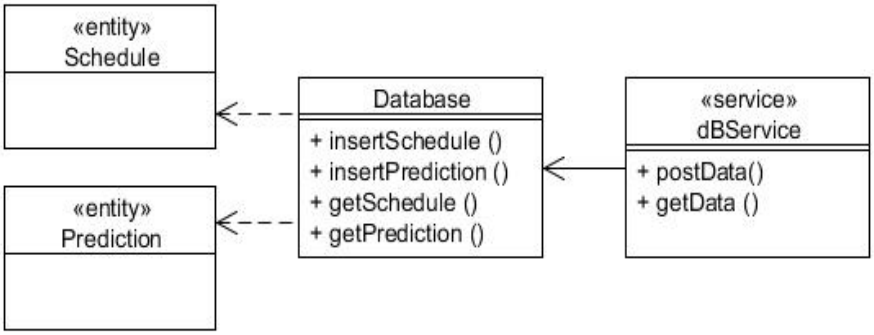



Figure 5.4: Database Access Class Diagram

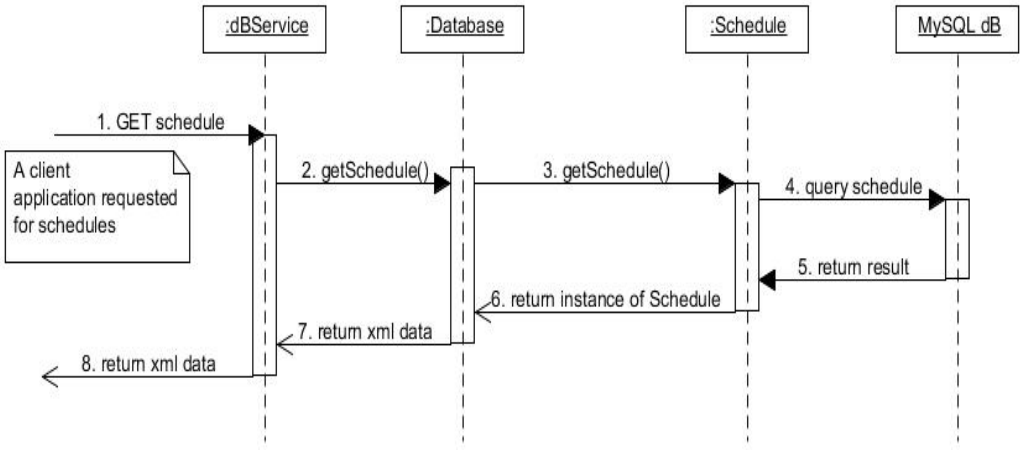



Figure 5.5: Insert Prediction Schedule Sequence Diagram

### Prediction Manager

Prediction Manager is used to manage all the predictions in the system.

The Manager class implement main () and order () methods. The main () method starts Monitor and Scheduler threads. The order () method uses ModelInterface object to order a model interface module to make a prediction.

The Monitor thread always read scheduled prediction from database and use the Manager order () method to order a model interface module.

The Scheduler thread schedule a prediction based on user defined configuration file that comes along the system.

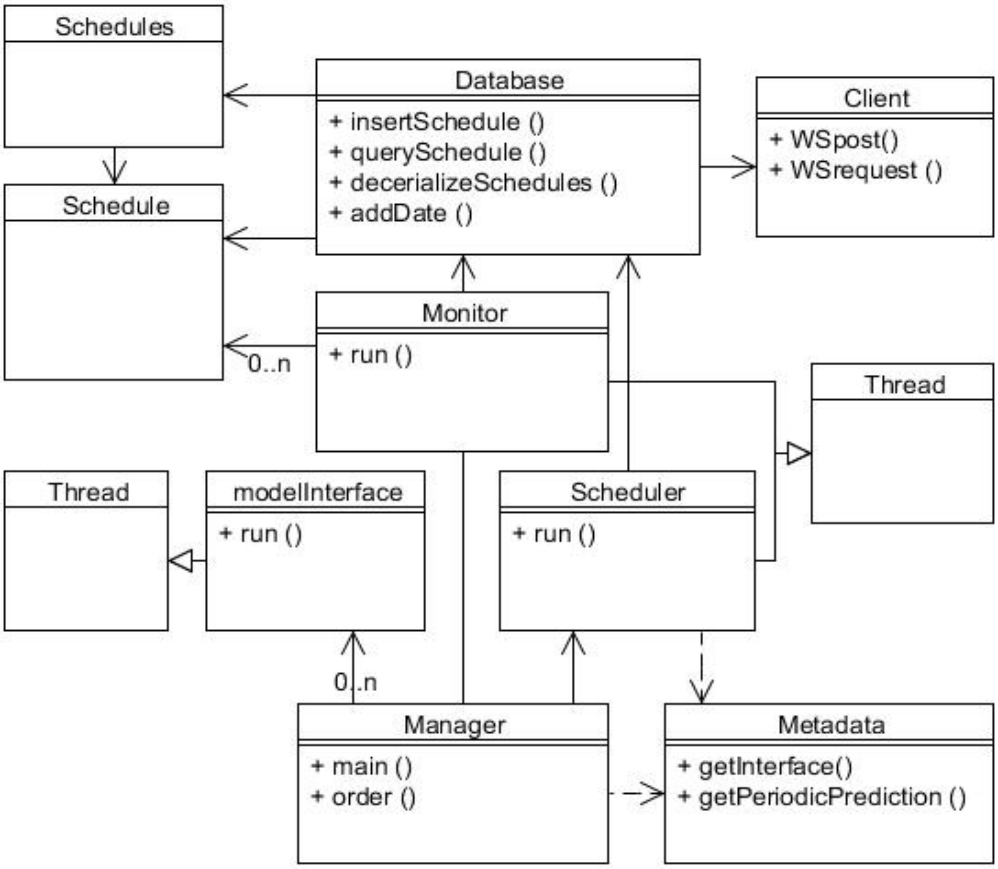



Figure 5.6: Prediction Manager Class Diagram

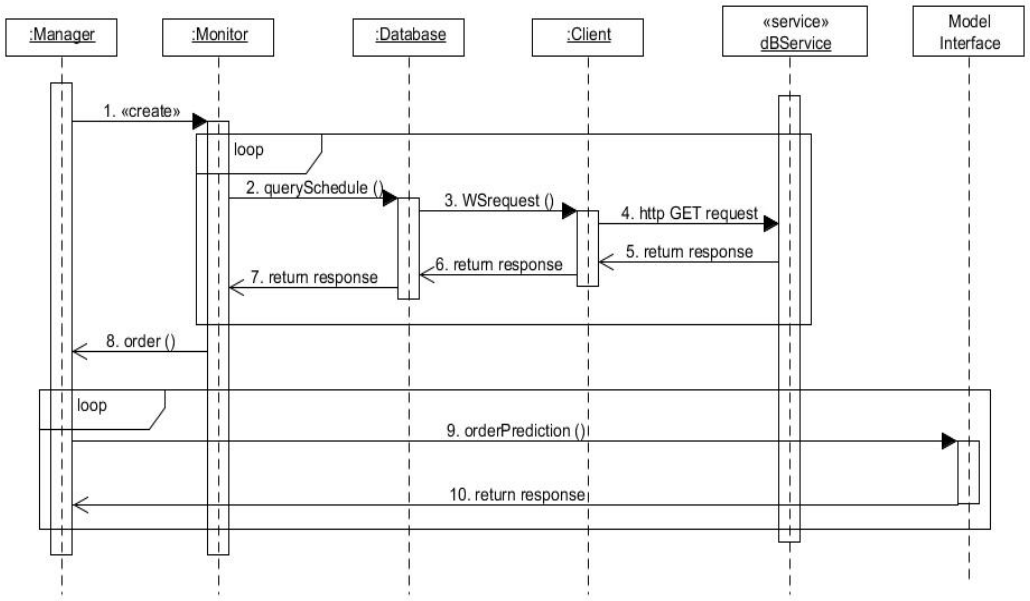



Figure 5.7: Prediction Manager Sequence Diagram

### STEM Interface

STEM interface performs input/output data processing and model execution of compartmental model. The Model Class implement main () and run () methods. The run () method executes the model.

The Parameters Class implements methods that input the required parameters to the model. The Prediction Class implements methods that insert prediction results to database and plot the prediction result. The Client Class implement post () method that performs as a web service client.

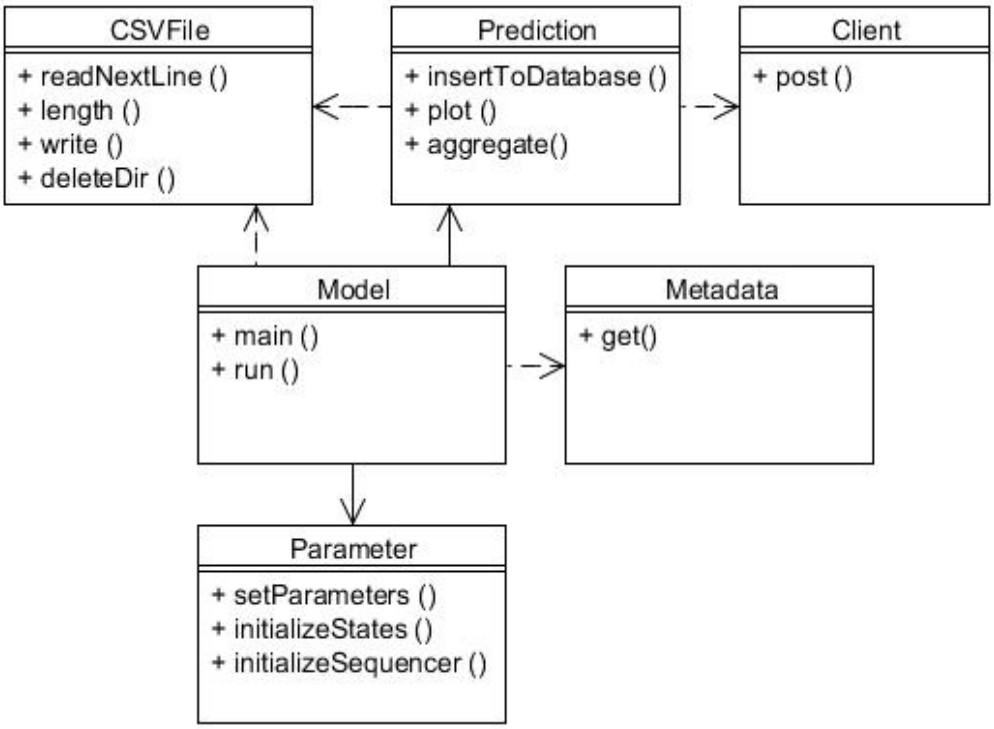



Figure 5.8: STEM Interface module Class Diagram

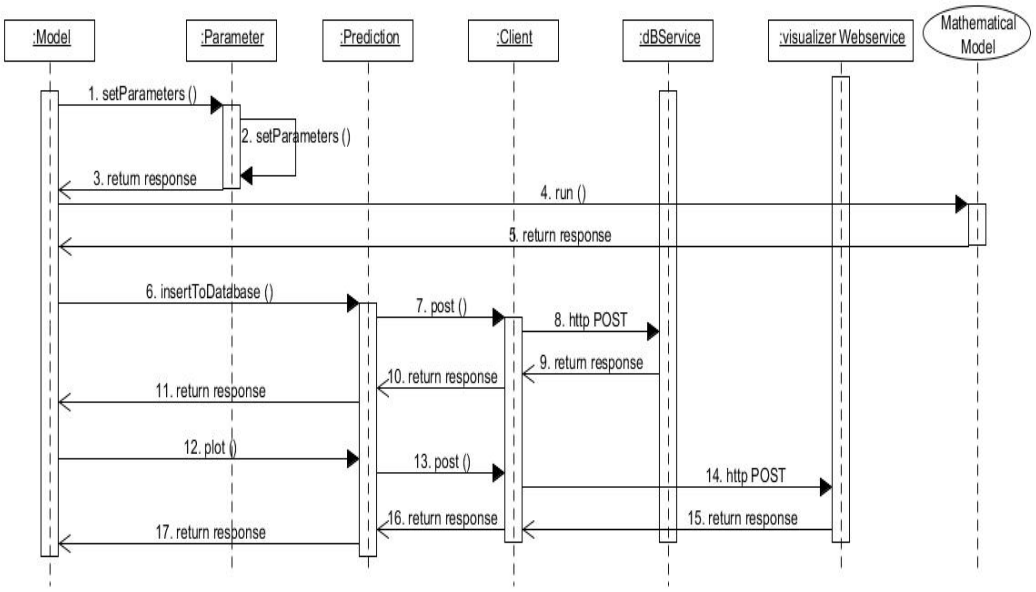



Figure 5.9: STEM Interface Sequence Diagram

### Bayesian Interface

Bayesian interface performs input/output data processing and model execution of the Bayesian model. The module has similar structure as STEM interface, except the internal operation of the methods.

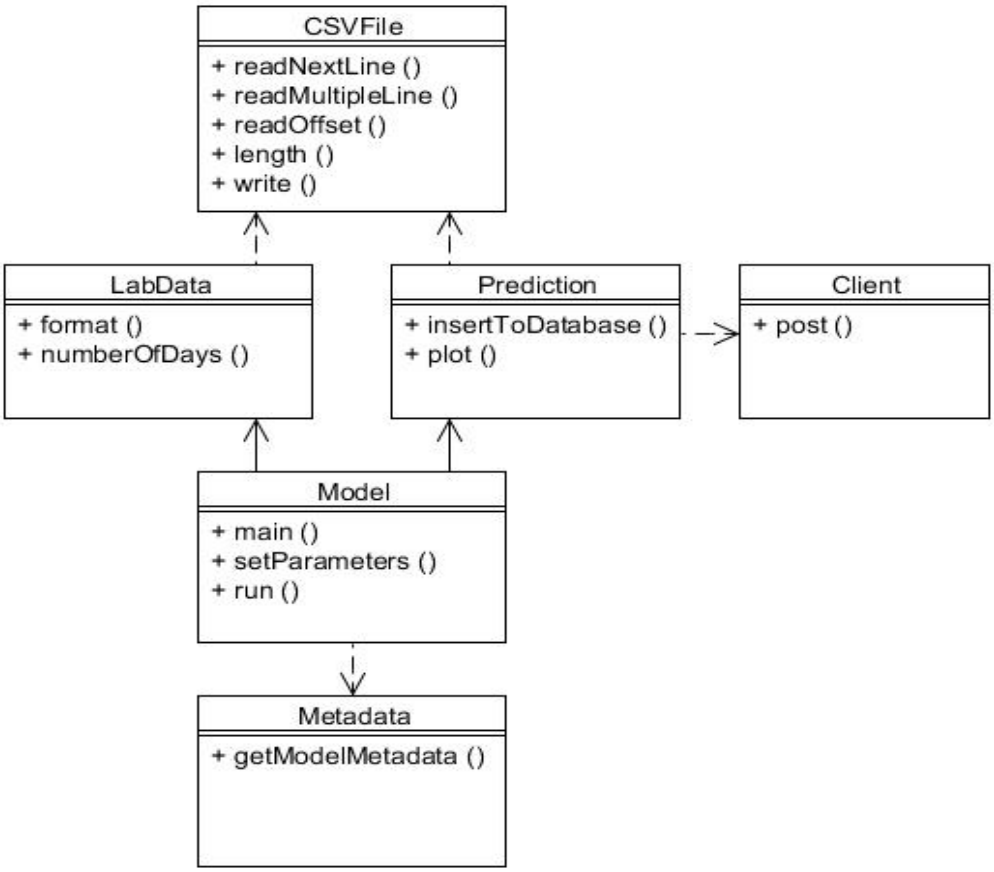



Figure 5.10: Bayesian Model Interface Class Diagram

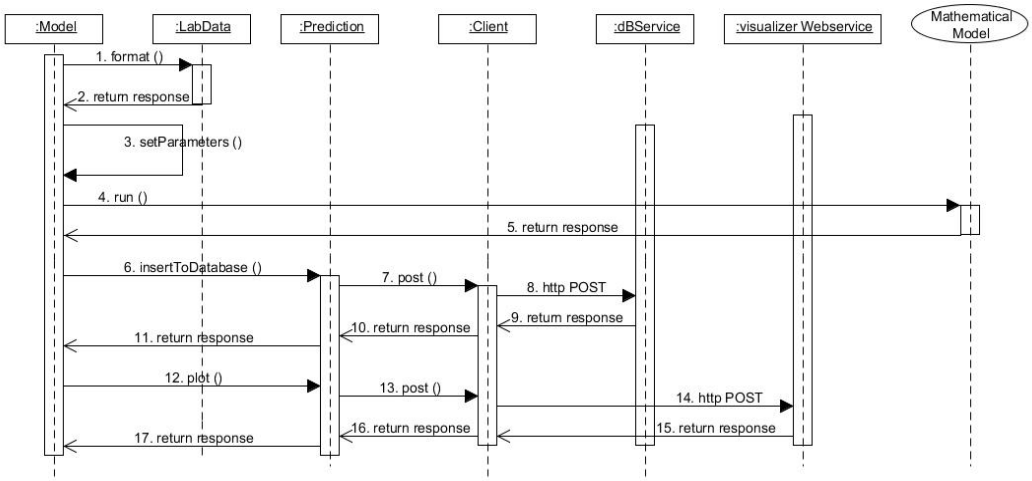



Figure 5.11: Bayesian Model Interface Sequence Diagram

### Visualization Manager

Visualization Manager generates maps and charts for visualization of prediction results. The visualizationService is JAX-RS resource class. The Map Class plot a map of the prediction result received from the web service interface.

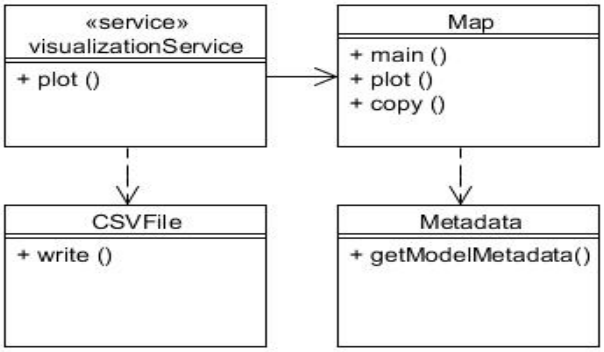



Figure 5.12: Visualization Manager Class Diagram

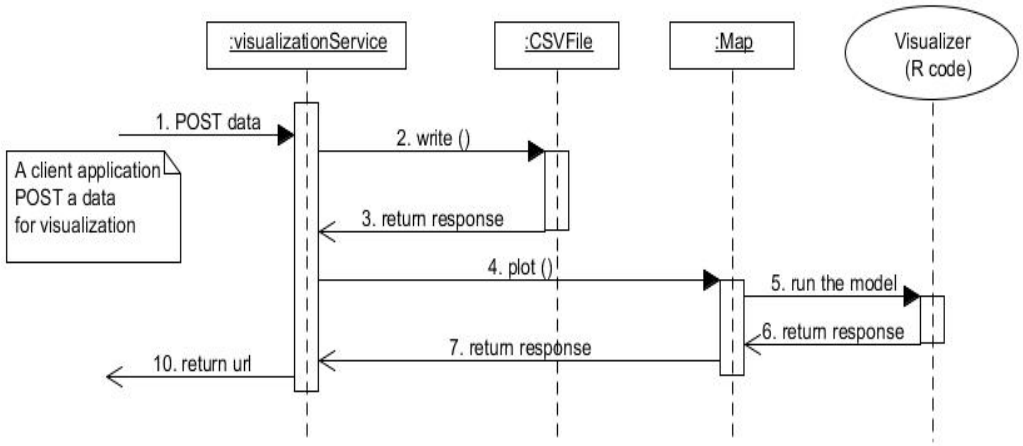



Figure 5.13: Visualization Manager Sequence Diagram

### Comparison Manager

Comparison Manager controls evaluation and comparison of mathematical models. The ComparisonService is JAX-RS resource class. The Manager Class implements order () method. The order () method runs a module that perform comparison/evaluation.

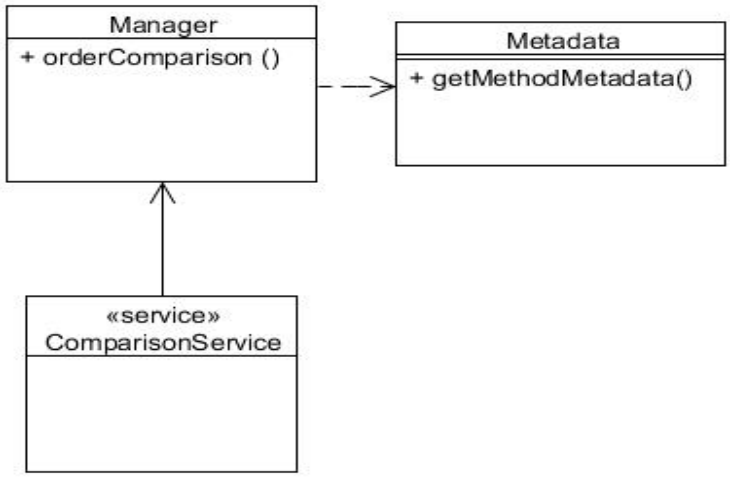



Figure 5.14: Comparison Manager Class Diagram

### Error Function

The Error Function performs NRMSE calculation. Function Class implements NRMSE () method that specifies the normalize root mean square error function. Error Class implements method to calculate error. Reference Class implements methods that read, aggregate and return the reference data. Prediction Class implements methods that read, aggregate and return the prediction data.

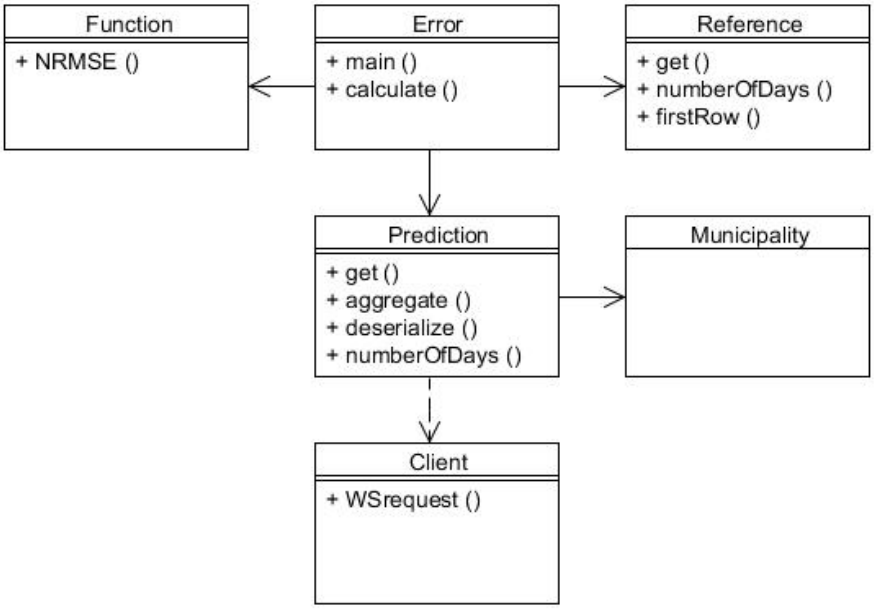



Figure 5.15: Error Function Class Diagram

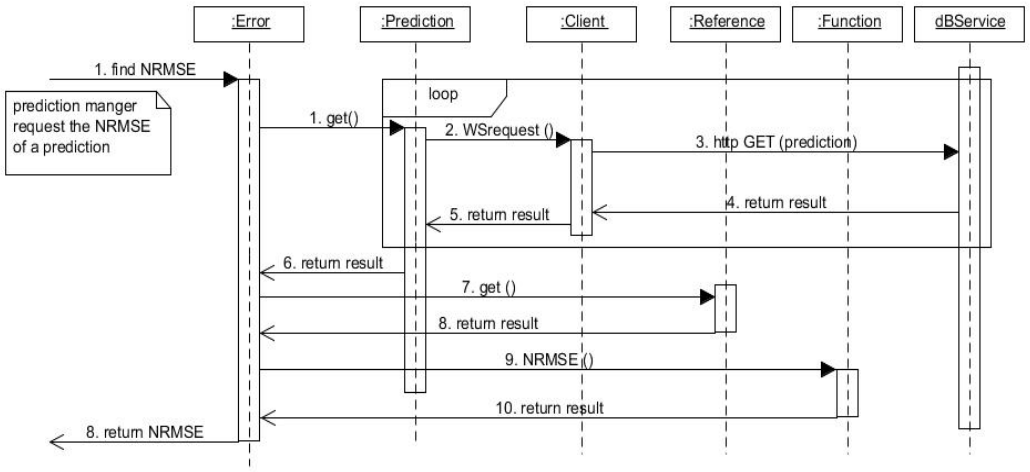



Figure 5.16: Error Function Sequence Diagram

## 5.6 Interface Design

The infectious disease prediction service comprises four user interfaces: Main, Comparison, Schedule and Data Upload. These interfaces will allow the users to access the services.

The Main interface is a visualization environment that is used to visualize spatio-temporal periodic prediction results. The interface has three integrated and synchronized views: periodic, and GIS. The periodic view displays periodic temporal prediction results on a table. The GIS view displays prediction results on a map.

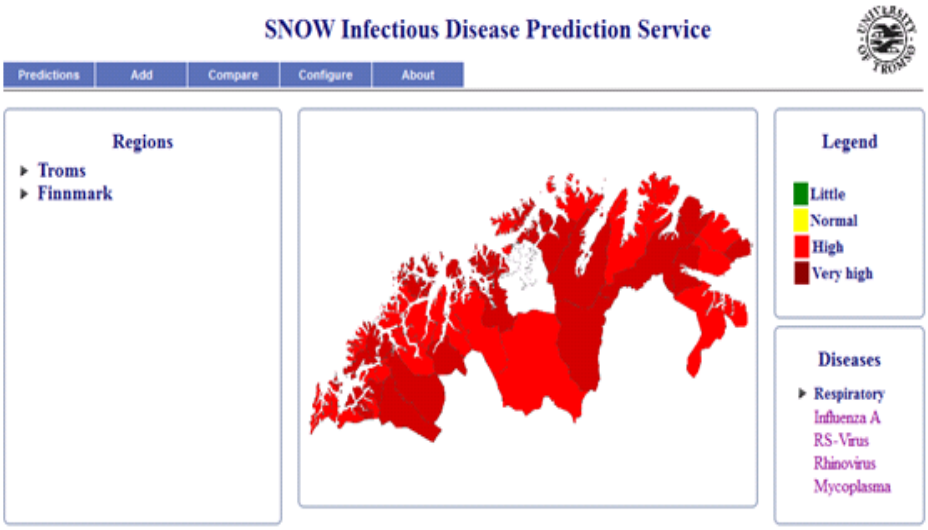



Figure 5.17: Screenshot of Main interface

Comparison interface allows Dr. Snow and Bernoulli to compare different mathematical models.

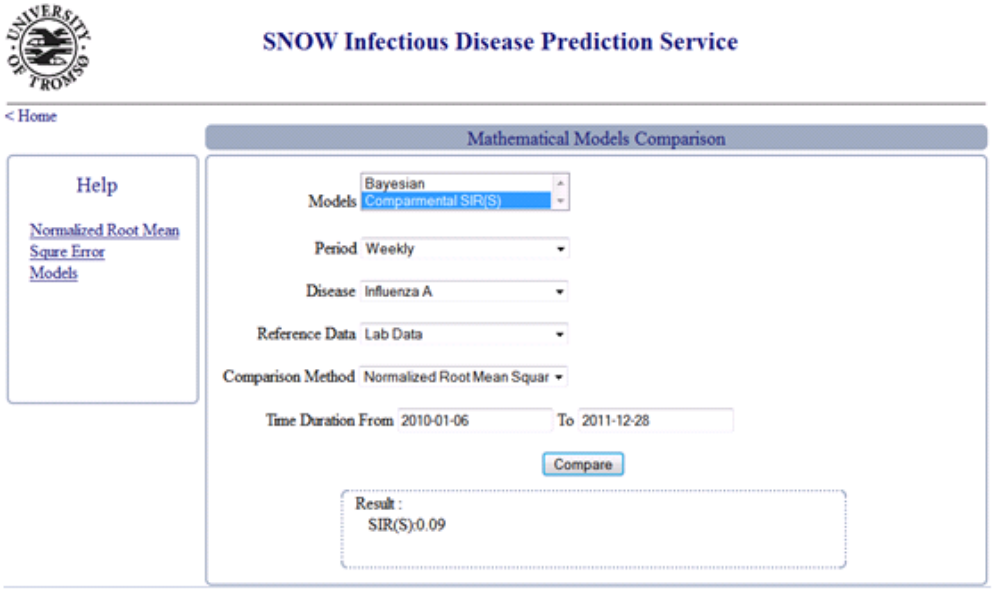



Figure 5.18: Screenshot of Comparison interface

Schedule interface allows Dr. Snow to schedule a prediction and access their scheduled prediction results.

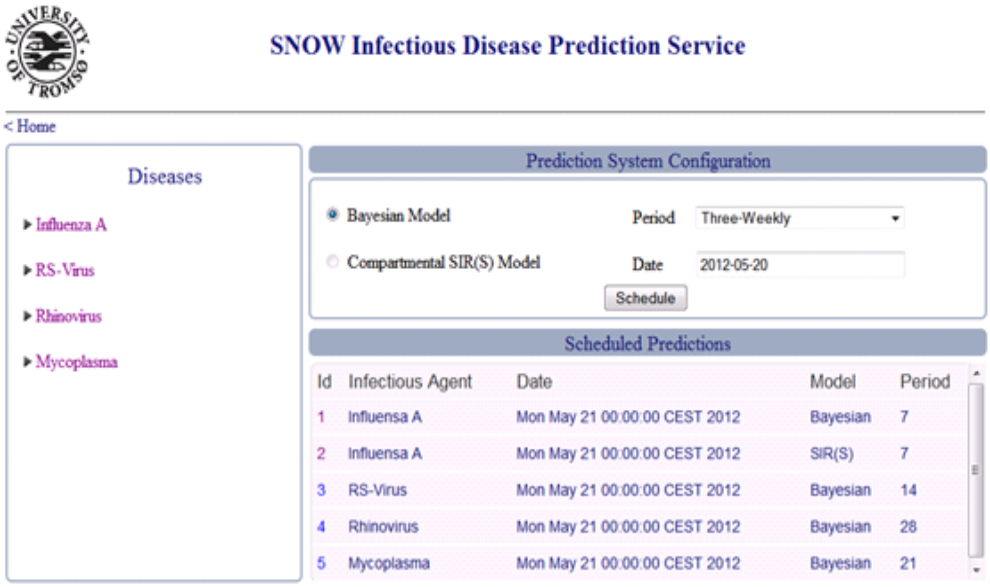



Figure 5.19: Screenshot of Schedule interface

Data Upload interface allows Dr. Snow and Bernoulli to interactively upload and display their own geospatial data files in csv format.

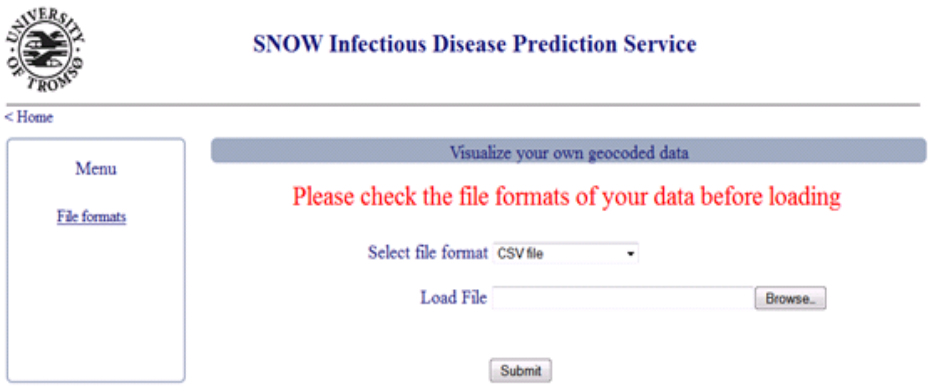



Figure 5.20: Screenshot of Data Upload interface

## 5.7 Summary

The chapter contains an overall description of the design considerations and followed by architectural, data, component and interfaces designs.

The architecture of the system is designed using three layered architecture combined

with SOA. The reasons behind decisions made during the design and decomposition descriptions are also discussed.

The data design contains a discussion about information domain and database design of the system. The component design is presented using UML class and sequence diagrams of each component in the system. The interface design comprises a brief description and screenshot of the four user interfaces: Main, Comparison, Schedule and Data Upload.

# Chapter 6

# Implementation and Testing

## 6.1 Introduction

This chapter describes the implementation details of the prediction service designed in Chapter 5. The implementation in general has been done in an iterative and incremental approach. The first section describes programming languages and technologies that have been used for the implementation of the project. Implementation description and code fragments of components on each layer are presented. Then, the chapter present a description of the system testing performed Finally, the chapter concludes with and requirement matrix and summary.

## 6.2 Programming language and Technologies

The data and business layer of the system is written in Java, only maps generating code of the visualization module is implemented in R. We decided to implement our code in Java because it is free, run on a wide range of operating systems, and the developer had better Java experience. The development was done in Eclipse environment as it is free and has many plug-ins.

The prediction service makes use of Java libraries such as Apache Xerces2 Java (Foundation 2010), Eclipselink (Clarke & Smith 2008), Jersey (Jersey 2008) and XStream (Gonzalez 2010).

The web client is made up of html, cascading style sheets (CSS), embedded JavaScript with AJAX capability and JSP at server side. We have used jQuery, JavaScript Library, for the Ajax and JavaScript.

### 6.2.1 Web Services

It is clear that the different system components must communicate with one another in order to provide a system that satisfies the user requirements. To establish well-defined interfaces for communication, we have used RESTful web service.

We chose to implement RESTful web service because it can be built with minimal tooling and the development is inexpensive with simple interface design (Pautasso et al. 2008) (ORACLE 2012). We have used JAX-RS (ORACLE 2012) implemented in Jersey (Jersey 2008). JAX-RS is a Java API that is designed to make it easy to develop applications that use the REST architecture.

### 6.2.2 Object-Relational Mapping

Java Persistence API (JPA) deals with the way relational data is mapped to Java objects ("persistent entities"), the way that these objects are stored in a relational database. We chose JPA as it is easy to implement and has better performance (Lascano 2008). We have used Eclipselink, which is the reference implementation of JPA and supports leading relational databases (Clarke & Smith 2008).

## 6.3 Data Layer

We chose MySQL database to store prediction schedule and results. Database access module provides web service interfaces (see Figure 6.1) for accessing the database.

```
//...
// process HTTP GET requests for prediction results and schedules
@GET
@Produces(MediaType.TEXT_PLAIN)
public String getData(@QueryParam("queryString") String queryString) throws java.text.ParseException {

    String [] queryArray=queryString.split(",");
    Query query=new Query();

    if (queryArray.length==4)
    {
        int id;

        //Find the schedule id for the prediction specified as as [infectious_agent,date,model, period]
        id=Integer.parseInt(query.getSchedule(queryArray[0], queryArray[1], queryArray[2], queryArray[3]));

        //return the prediction
        return(query.getPrediction(id));
    }
    else
        //return schedules for a given date
        return(query.getSchedule("",queryArray[0],"",""));
}

//...
```

Figure 6.1: JAX-RS resource class code fragment for querying schedule and prediction result

The databases access module uses JPA to query and insert predictions and prediction schedules. Figure 6.2 shows a code fragment that uses JPA entity class to insert a schedule to database.

```
//...
//Object of entity class Schedule
Schedule schedule=new Schedule();
SimpleDateFormat dateFormat= new SimpleDateFormat("yyyy-MM-dd");
Date scheduleDate=(Date)dateFormat.parse(date);

//set parmaeters of the Schedule
schedule.setInfectous_agent(infectiousAgent);
schedule.setDate(scheduleDate);
schedule.setModel(model);
schedule.setPeriod(period);
try
{
    EntityManagerFactory emf = Persistence.createEntityManagerFactory("database-access");
    EntityManager em=emf.createEntityManager();
    em.getTransaction().begin();      //begin transaction
    em.persist(schedule);
    em.getTransaction().commit(); //commit transaction
    em.close();
    return true;
}
//...
```

Figure 6.2: JPA code fragment for inserting new schedule into database

As mentioned in section 2.6, the Snow system stores the laboratory data in a database running on a server at UiT. The Snow interface module, which also runs on the same server, queries the database using JPA. The module parses the laboratory data using Apache Xerces2 Java Parser and prepares it into a csv format shown in Figure 6.3.

| iteration | time | NO-19-01 | NO-19-02 | NO-19-11 | NO-19-13 | NO-19-15 | NO-19-17 | NO-19-19 | NO-19-20 | NO-19-22 | NO-19-23 | NO-19-24 | NO-19-25 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 126 | Wed 14 Oct 09 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 127 | Wed 21 Oct 09 | 2 | 9 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 3 | 0 |
| 128 | Wed 28 Oct 09 | 6 | 25 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 16 | 1 |
| 129 | Wed 04 Nov 09 | 41 | 70 | 1 | 4 | 0 | 0 | 1 | 0 | 6 | 0 | 14 | 8 |
| 130 | Wed 11 Nov 09 | 11 | 83 | 1 | 1 | 0 | 0 | 1 | 1 | 7 | 1 | 15 | 0 |
| 131 | Wed 18 Nov 09 | 7 | 35 | 0 | 0 | 0 | 0 | 2 | 1 | 8 | 0 | 4 | 8 |
| 132 | Wed 25 Nov 09 | 1 | 47 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 2 |
| 133 | Wed 02 Dec 09 | 1 | 28 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 134 | Wed 09 Dec 09 | 0 | 12 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 135 | Wed 16 Dec 09 | 0 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 136 | Wed 23 Dec 09 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Figure 6.3: A csv file fragment processed by the Snow interface (Note: municipalities are defined using ISO 3166-2 code)

## 6.4 Business Layer

### Prediction Manager

As discussed in the previous chapter, the Prediction Manager manages all predictions in the system. Every day, the manager uses the database access module web service interface to monitor scheduled predictions.

```
<?xml version="1.0" encoding="UTF-8"?>
<schedules>
    <schedule>
        <id>20</id>
        <infectious_agent>Influenza A</infectious_agent>
        <model>SIRS</model>
        <date>Tue May 08 00:00:00 CEST 2012</date>
        <period>Weekly</period>
    </schedule>
</schedules>
```

Figure 6.4: Sample prediction schedule

```
public ArrayList<Schedule> decerializeSchedulesXML(String record)
{
    Schedules schedule=new Schedules();

    try {
        String dir=new File(".."+separator).getCanonicalPath().toString();
        String workspace=separator+"Prediction Service"+separator+"Workspace"+separator; //workspace

        FileWriter writer = new FileWriter(dir+workspace+"schedules.xml");
        writer.append(record);
        writer.flush();
        writer.close();

        XStream xs = new XStream(new DomDriver());

        FileInputStream xmlFile = new FileInputStream(dir+workspace+"schedules.xml");
        xs.addImplicitCollection(Schedules.class,"schedule",Schedule.class);
        xs.alias("schedule", Schedule.class);
        xs.alias("schedules", Schedules.class);
        xs.fromXML(xmlFile,schedule);    // deserialize the schedules.xml file

        //return the object containing the decerialized data
        return schedule.schedule;
    }
    //...
```

Figure 6.5: Code fragment to deserialize the schedule retrieved from database

Then, the prediction manager deserializes (XStream is used) the XML format schedule received from the database access web service and order a model interface module to

make prediction.

The model Interface performs input data formatting, set model parameters and executes the model. Then, the interface reads the prediction results and inserts the result to the database, in xml format. Finally, the interface visualizes the result.

```
<?xml version="1.0" encoding="UTF-8"?>
<prediction period="7" date="2012-04-21" model="Bayesian">
    <area code="1901">
        <result>18.72139</result>
    </area>
    <area code="1902">
        <result>27.00325</result>
    </area>
    <area code="1911">
        <result>1.29966</result>
    </area>
    <area code="1913">
        <result>1.71045</result>
    </area>
    ...
</prediction>
```

Figure 6.6: A fragment of sample prediction result in database

### Comparison Manager

The Comparison Manager receives model evaluation requests through its web service interfaces. The module uses comparison techniques, implemented as independent module, to do the evaluation. In this thesis we have developed a module that implements NRMSE (NRMSE is discussed in section 2.5.3).

```
//...
//Read First period values
referenceSum=reference.getIncidence(referenceData,"reference");
predictioSum=prediction.getIncidence(predictionData,"prediction");

//Calculate mean square error for the first period= (currentPeriodPrediction-currentPeriodReferrence)2
meanSquareError=meanSquareError+(Math.pow((predictioSum-referenceSum),2));

//initialize the max and min reference sums with the first period values
maxReferenceSum=referenceSum;
minReferenceSum=referenceSum;

for(int i=0;i<period;i++)
{
    //read the aggregated values for the current period
    referenceSum=reference.getIncidence(referenceData,"reference");
    predictioSum=prediction.getIncidence(predictionData,"prediction");

    //Calculate mean square error
    meanSquareError=meanSquareError+(Math.pow((predictioSum-referenceSum),2));

    //find the min and max reference sum
    if(referenceSum>maxReferenceSum) maxReferenceSum=referenceSum;
    if(referenceSum<minReferenceSum) minReferenceSum=referenceSum;
}

//the Root Mean Square Error
rootMeanSquareError=Math.sqrt(meanSquareError/period);

//the NRMSE
NormalizedRootMeanSquareError=rootMeanSquareError/(maxReferenceSum-minReferenceSum);
//...
```

Figure 6.7: Code fragment of a method that implements NRMSE

### Visualization Manager

STEM had a plug-in for the map of Norway at counties level and we developed a new plugins of Norway map with municipality level resolution. The new map plug-in along with the logger and external play-back features of STEM can be used to visualize prediction results. But, we found it flexible to visualize prediction results by using a code written in R (Venables & Smith 2012). The R code uses Norwegian municipality level map file (NOR_adm2.RData) available at (http://gadm.org/). The visualization module provides standard web service interfaces to receive requests from the web client and model interfaces.

## 6.5 Presentation Layer

The presentation layer is a web client made up of static and dynamic html pages including cascading style sheets (CSS), maps, tables, embedded JavaScript with AJAX

Table 6.1: Requirement matrix

| Requirements | Sub systems | Status |
|---|---|---|
| Get lab data | Data source manager and Snow interface | Implemented |
| Periodic predictions | All | Implemented |
| Configurations | Prediction manager | Implemented |
| Schedule predictions | Web client and Database access | Implemented |
| Evaluate Mathematical Models | Comparison manager, Database access, web client and Error function | Implemented |
| Visualize External Data | Visualization Manager and Web client | Implemented |
| Query Prediction Results | Database Access | Implemented |

capability and JSP at server side.

## 6.6 Testing

The system has been tested aiming at evaluating functionalities of the software system. Following completion of each sub task in the iteration we did unit testing, where we tested each module to check whether the individual modules are working properly and applied the test result into the module when we find error.

Finally, Interface testing between individual software modules has been done to check whether individual modules are communicating properly. There was no chance involving real users for controlled lab experiment on the visualization. Thus, only the developer did user interface testing.

After the development, the system has been used for the retrospective prediction of various diseases (see chapter 7, page 78). As the system is designed for real time prediction, we had to customize the system somehow to make it work in the past.

## 6.7 Requirements Matrix

Table 6.1 presents the links between the requirements and the project design and implementation.

## 6.8 Summary

For the implementation of the design presented in chapter 5, mainly, we have used Java programming and technologies such as JAX-RS and JPA. R is used to write the code that creates maps corresponding to prediction results. The presentation layer is implemented as a web client based on current web technologies. Implementation description and code fragments of components on each layer are presented. Finally, a requirement matrix is presented.

# Chapter 7

# Mathematical Models and Evaluation

## 7.1 Introduction

This chapter describes the method and results of the SIR(S) Influenza A model and evaluation of the Bayesian and SIR(S) models. First we present justifications that Influenza A can be studied using SIR(S) and the method used. Then, the method used to evaluate various diseases using the Bayesian model is presented. Finally, the chapter concludes with the results and discussion of the two models.

## 7.2 Influenza A SIR(S) Model

Mathematical models have been used to understand the spatial-temporal transmission dynamics of influenza (A. Rvachev & Longini Jr. 1985) (Aguirre & Gonzalez 1992) (Edlund et al. 2011a) (Edlund et al. 2011b). SIR compartmental model has been used as a basis for influenza models. Detailed description about compartmental models can be found in section 2.4.2.

Individuals recovered from Influenza typically, at least partially, become susceptible to new variants of influenza virus within a few years (White & Fenner 1994). Thus, Influenza A can be studied with SIRS compartmental model representing the passage

of individuals between Susceptible ($S$), Infectious ($I$), and Recover ($R$) states and eventually returning to S state as immunity is lost. Since influenza has a short latent period, for simplicity, we do not explicitly model the exposed state.

Recent studies have shown that seasonal variation in influenza (Shaman & Kohn 2009) incidence can be explained by small changes in transmission rate triggered by changes in temperature and relative humidity (Lowen et al. 2007) (Shaman et al. 2010).

In this section our goal is to model Influenza A SIR(S) model covering the study area, with a seasonally modulated transmission coefficient and air transportation model between the municipalities. We chose Influenza A because it is a well studied infectious disease with well understood bounds on the disease parameters and the size of Influenza A incidence every year.

## Data

The modeling is based on the Influenza A data discussed in section 3.2.2. The number of test results from the laboratory represents only a fraction of the total incidence. We assume that the lab results represent a constant proportion of actual disease incidence, but the actual (average) testing fraction is unknown. Methods for finding the reporting fraction is reported in (Edlund et al. 2011b) (Edlund et al. 2011a), but the methods are time intensive. To simplify the modeling, based on the result from (Edlund et al. 2011b), we made an assumption that the reporting fraction is 3%.

The rate of immigration (through birth and otherwise) and emigration (through death and otherwise) were estimated using data from the Norwegian statistics bureau (Statistics Norway 2011) and considered to be constant across all the municipalities in northern Norway.

## Method

As discussed in section 2.4.2 each compartment of a SIR(S) model is defined as spatio-temporal differential equation. Runge Kutta Fehlberg method is used to integrate the differential equations determining ($S$), ($I$), and ($R$) . In this thesis we chose the Runge Kutta Fehlberg method, as it takes advantage of adaptive step-size algorithm. The integration results are much more accurate.

If the model parameters are constant in time, the model will not produce seasonal variation of the disease. To overcome this limitation, STEM has models containing a "forcing term" that seasonally modulates the transmission coefficient $\beta(t)$ with a period function (Edlund et al. 2011a). The forcing term includes a modulation exponent, $\lambda$, to provide independent control of the peak transmission "seasonal duty cycle" expressed in Equation 7.1.

$$\beta(t) = \beta_0[(1.0 - a) + a|\sin(\omega t + \varphi)|^{\lambda}] \tag{7.1}$$

Where: $\varphi$ = Modulation phase shift

$a$ = modulation floor

$\omega$ = Angular frequency

For our model, the geographic transmission dynamics of influenza virus caused by transportation can be an important factor in understanding the evolution of influenza in space and time. Thus, we have included interactions between adjacent locations and locations connected by air transport.

**Model Fitting**

To optimize the parameters $\beta$, $\delta_0$, $\lambda$, $a$, and $\psi$, we fit two years and six months data using literature values (Edlund et al. 2011a) for the recovery rate, $\gamma \approx 0.1$ (10 days infection period) and immunity loss rate, $\alpha \approx 0.0007$ (immunity lasting about 4 years). We fit the model parameters using the Nelder-Simplex algorithm implemented in the automated experiment feature of STEM (see section 2.5.3).

Using the best fit, we then ran the model for the next two years and measured the NRMSE against the laboratory influenza data.

## 7.3 Bayesian Model

A detailed description about the Bayesian model can be found in section 2.4.3. It is a spatio-temporal general purpose infectious disease prediction model. The model accounts for the spatial transmission caused by air transportation and interaction between

Table 7.1: NRMSE values of Bayesian model weekly and monthly (defined as 4 weeks) predictions of various diseases

| Diseases | Weekly | Monthly |
|---|---|---|
| Influenza A | 0.0721 | 0.1713 |
| RS-virus | 0.1299 | 0.2437 |
| Rhinovirus | 0.1498 | 0.2545 |
| Norovirus | 0.1243 | 0.1863 |
| Mycoplasma pneumoniae | 0.1424 | 0.1735 |

adjacent municipalities. We have applied the model to various infectious diseases such as Influenza A, Rhinovirus, Mycoplasma pneumoniae, RS-virus and Norovirus.

### Method

For a weekly prediction, Influenza A, RS-virus, and Norovirus datasets until Jan 2010 and Mycoplasma pneumoniae and Rhinovirus datasets until Oct 2010 are weekly aggregated for each municipality. Then, we made weekly prediction for all the next weeks until Apr 2012 by recurrently appending a weekly aggregated data to all the previous dataset in the series. Finally, the NRMSE is calculated for each prediction.

We followed similar procedure for a monthly (four weekly) predictions, except the datasets are four weekly aggregated.

## 7.4 Results and Discussion

We made weekly and monthly predictions of Influenza A, RS-virus and Norovirus for around two years and Rhinovirus for around one and half year. We also did one and half, and two years weekly and monthly prediction of Mycoplasma pneumoniae respectively. Table 7.1 shows the NRMSE values of the Bayesian model weekly and monthly prediction of each disease. Note that the smaller the NRMSE value the better the prediction fit to the reference data. In both cases, the Influenza A predictions appeared to be best predicted. The model's goodness-of-fit decreased for the monthly prediction.

From the SIR(S) compartmental model fitting a minimum NRMSE, 0.1627, was found

using the following parameter values: $\beta = 0.3$, $\delta_0 = 3.28$, $\lambda = 0.92$, $a = 0.25 = 0.25$, and $\psi = 2.15$. Figure 7.1 shows the first three years of actual weekly Influenza A laboratory cases (summed over municipalities) as well as best fit model.

During the 2009 H1N1 pandemic the Influenza A has peaked twice and there was cases many folds larger than other seasons. In addition to, there was massive, 45% of the total population (Guzman Herrador et al. 2012), vaccination during the same season. Thus, it can be an explanation why the fit has large NRMSE, as these was not considered in the fitting.

The basic reproductive number $R_0$ for Influenza A estimated from the model fit is about 3.0, which is within the range found in other literature (Andreasen et al. 2008).

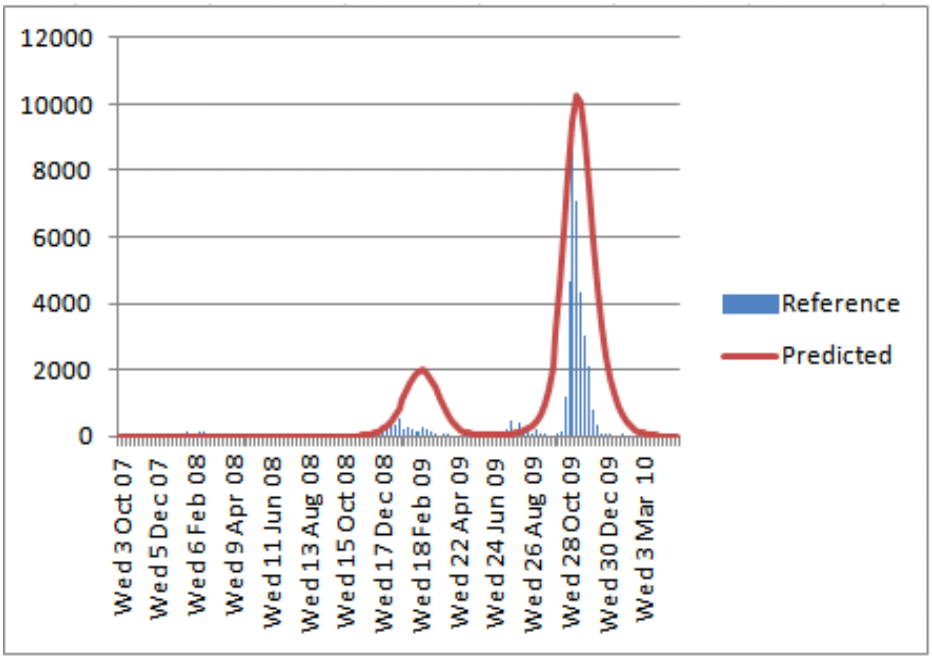



Figure 7.1: Weekly Influenza A cases (summed over all locations) for fitted model (red) and actual Influenza A data (blue)

Using the best fitting parameters of SIR(S) model, we made a prediction for the next two years and found an NRMSE value of 0.1333 for weekly and 0.15 for monthly aggregated predictions. Figure 7.2 shows two years of actual weekly Influenza A cases (summed over municipalities) as well as Bayesian and SIR(S) models weekly prediction.

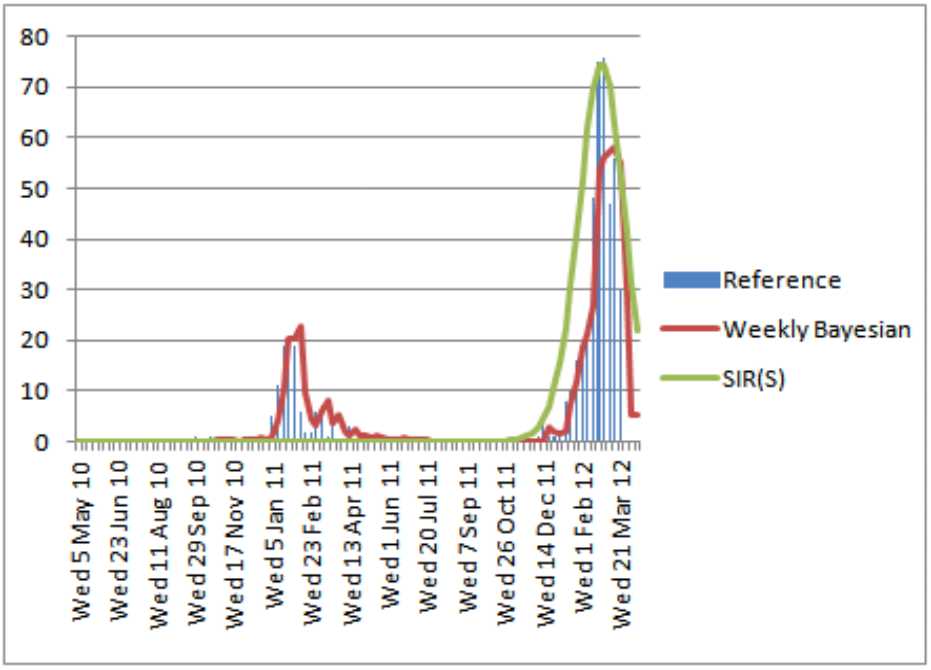



Figure 7.2: Weekly Influenza A cases (summed over all locations) for actual Influenza A (blue), Bayesian model (red) and SIR(S) model (green)

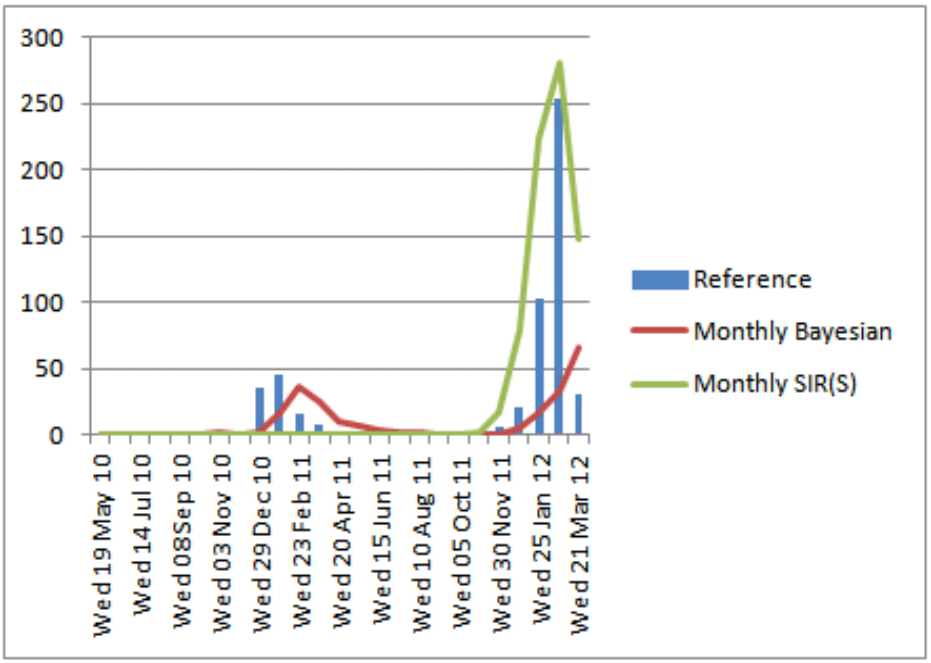



Figure 7.3: Monthly Influenza A cases (summed over all locations) for actual Influenza A (blue), Bayesian (red) and SIR(S) (green) models

Figure 7.3 shows two years of actual monthly Influenza A cases (summed over municipalities) as well as Bayesian and SIR(S) models monthly prediction.

From the NRMSE values per se it appear the Bayesian model's weekly prediction has better goodness-of-fit over the seasonal SIR(S), and the opposite is true in case of monthly prediction. From the Figures 7.2 and 7.3 the SIR(S) model did not catch the first season, while the Bayesian model caught both seasons.

# Chapter 8

# Results and Discussion

## 8.1 Introduction

This chapter presents major findings of the thesis and discussion of the findings. The chapter begins with the findings of the prediction service framework and followed by the results of the mathematical models in the thesis. Importance of the prediction service and the prediction service framework is discussed. Then, we did a comparison with similar studies. Finally, the chapter concludes with limitations of the thesis.

## 8.2 Prediction Service Framework

Infectious disease prediction models are modeled at different level of assumptions and knowledge that defines dynamics of disease spread. Thus, potentially each model requires a unique input datasets and no model will ever be completely accurate.

Information system that integrates epidemiological models from input data to the level of prediction and visualization could enable better and timely analysis, and potentially improved outbreak control.

This thesis introduced a technical framework designed for creating an integrated infectious disease prediction service, which is implemented for north Norway. The framework facilitates the process of integrating various models requiring different input and

output formats. We have tested the integration using two (Bayesian and Compartmental) mathematical models that demonstrate the effectiveness of the architecture integrating new model.

### 8.2.1 Data Source

Currently, the Snow diseases surveillance system is the only data source. However, the modular design supports the addition of new data sources as each data source has an independent interface module. The prediction system cache weekly aggregated historical laboratory data necessary for making predictions, which avoids having to re-request all the historical counts required by the prediction models. But if there would be any risk of caching the historic data in one location, the interface module enables to apply distributed storage.

### 8.2.2 Prediction

The system has a separate configurable model interface module (Bayesian and STEM Interface) that executes a model. Since the operation of input/output data processing and model execution of each model is self-contained and strictly defined in the interface module, integration of new models is straightforward.

Similar to weather forecasting systems, the system periodically predicts the spread of diseases that are defined in configuration files that comes with the system. The configuration files are in text format that can be modified by an expert user to set periodic prediction parameters, such as the model and data source for a particular disease.

Expert users (i.e. Epidemiologists) can schedule one time prediction of a particular disease and get the results using the interface shown in Figure 5.19.

### 8.2.3 Visualization

Visualization facilitates access to prediction results and support decision making. We designed map-based interfaces (see Figure 5.17) to visualize the spatio-temporal prediction results.

The system allows users to interactively upload and display their own geospatial data files as shown in Figure 5.20. Data files are supplied in csv file format with values corresponding to the municipality name and disease counts. The visualization manager utilizes a code written in R to create a color coded map corresponding to the uploaded data.

## 8.3 Evaluation of Mathematical Models

To facilitate the evaluation of the different models, the framework incorporated an evaluation module, Error Function, which implements NRMSE. Access to the module is achieved through the comparison manager web service interface. When a user request for NRMSE calculation from the web client, the comparison manager executes the error function to find the error value (see Figure 5.18).

Since all models use the same laboratory data, comparing the prediction results against the reference data can provide insights into the accuracy of a certain model for a given disease. The module allows assessment of the models goodness-of-fit to the data they are used on. The modular design also supports the addition of more comparison modules.

Bayesian and SIR(S) compartmental models are the two models that are integrated in the system. The first model is a general disease model, while the second model is an Influenza A model. To evaluate the performance of the models we have calculated the NRMSE of the models predictions against the laboratory dataset retrospectively. The results and discussion of the Influenza A model and performance evaluation of the Bayesian model for various diseases can be found in the Chapter 7. Here we present the result briefly.

### 8.3.1 Bayesian Model

We have applied the Bayesian model to various infectious diseases retrospective. The NRMSE values of the model predictions of various diseases are in range of $(0.0721, 0.1498)$ for the weekly predictions and $(0.1713, 0.2545)$ for the monthly predictions. Compared to the other diseases, the Influenza A predictions of the Bayesian model appeared to

be best predicted. For all the diseases, the model's goodness-of-fit decreased for the monthly prediction.

### 8.3.2 Compartmental Model

We fit various parameters using two years and six months data. Using the best fit, we then ran the model for the next two years and found a NRMSE value of 0.1333 against the Influenza A data.

From the NRMSE values per se it appear the Bayesian model's weekly prediction has better goodness-of-fit over the seasonal SIR(S), and the opposite is true in case of monthly prediction.

## 8.4 Importance of the Prediction Service

Public health preparedness and decision making concerns raised by current infectious disease threats has increased the need of epidemiological modeling in analyzing how infectious diseases will spread and its effective control mechanisms. The prediction service is designed to enable public health officials, policy makers, and the public to examine the spread of infectious diseases in a community.

In infectious disease prediction there is a trade-off between the specificity[1] of predictions (in space and time) and how many days ahead the predictions are made. In general, long-term predictions are the least specific, while providing enough lead-time to respond for the warning. Such predictions enable to prepare for disease outbreaks intervention planning.

On the other hand, systems based on early detection of cases provide highly specific information, but allow little time to respond for the warning. Such prediction information could have great value for healthcare professionals (i.e. GPs) daily clinical care and the public to deal with recently happening disease outbreak as weather forecasts could be important in our clothing choice or trip plan.

Our prediction service makes prediction from one to four weeks ahead. We set the

[1]Specificity measures the proportion of outbreaks which are correctly identified

prediction resolution to one week as the data source is weekly aggregated. Similarly, the maximum prediction length is set to four weeks because the Bayesian model needs more historical data for long term predictions.

The prediction service is designed in a way that the prediction results can be accessed through the web service interfaces. Thus, decision support systems (DSS) can use the prediction information in supporting complex and difficult decision-making. For example, the prediction results could be important to the disease query engine described in (Leer et al. 2012), which is basically a symptoms to disease matching engine. The engine uses infectious diseases cases from the Snow system as an indicator for the current disease status of the population, before applying a hierarchical Bayesian model to generate the probability for each disease. The prediction results can be a data source for such information, especially at the beginning of outbreaks as patients start having symptoms before GPs start sending their patients lab sample test to laboratories.

## 8.5 Comparison with Similar Studies

From the literature review of infectious disease detection and prediction systems, in section 2.5, a number of detection systems and simulation tools are reviewed. But, we came across only one prediction systems, SIMID (Villa et al. 2011).

Looking at the tremendous progress in mathematical modeling, the availability of detection systems and simulation tools, one could ask why we don't have comparable number of prediction systems. Even though detection systems also have a prediction power with little time before outbreak, why the prediction is usually not reported? Perhaps, studies are required to answer these and similar questions.

Currently, SIMID utilizes a network model to simulate Influenza outbreaks (Villa et al. 2011); while our system utilizes two models. SIMID and our system visualize the spatio-temporal simulation results using a map-based interface. SIMID does not have model evaluation while our system has evaluation mechanism with a capability to integrate new mechanisms.

Our system provides continuous prediction as weather forecasting systems; it also allows user to schedule a prediction. Whereas, SIMID implemented only prediction scheduling.

## 8.6 Limitations

The requirement gathering process involved only three individuals that represent the potential user groups. There was no chance involving these individuals for controlled lab experiment on the visualization. However, we tried to compensate the limited number of user participation by using personas. The personas we have created contain little number of personal details. Thus, how much we tried to avoid, there was a risk that personas could turn into a generic user instead of a precise design target.

As the data source is weekly aggregated, the system predict and visualize data in weekly resolution, which would delay the public health responses when there is a sudden change of disease activities, especially in the rising phase and during the evolving phase of an epidemic or pandemic situation.

In the SIR(S) model, to simplify the modeling (see section 7.2, Page 78), we made an assumption that the reporting fraction of the laboratory data as 3%. Our assumption was based on a research result for Israel Influenza like Illness diagnosis data, which may not be true for our data. Thus, a study is required to find the laboratory reporting fraction.

# Chapter 9

# Conclusion and Future Work

## 9.1 Conclusion

Epidemiological modeling enables to study how infectious diseases spread and evaluating possible strategies applied to control an outbreak. This thesis mainly focused on construction of an integrated infectious disease prediction service that predicts and visualizes prediction results in time and space.

The framework was designed using a three layered architecture combined with SOA, where modules on business and data layer expose their functionality as services.

The architecture enables to integrate new mathematical models, which is demonstrated using Bayesian general disease and seasonal SIR(S) compartmental Influenza A models.

In the following paragraphs, inferences are made about the significance of the thesis findings in relation to the research questions.

*Question 1: How can we construct a generic infectious disease prediction service framework that enables integration of new mathematical models?*

In this thesis the design and implementation of an extensible and scalable framework is described. The main anticipated change is integration of new mathematical models. As the model interface is separate from the main logic of the system, new model integration only requires addition of interface module that performs input/output data processing

and execution of the model using other services of the system. Only small changes in the client application will be required. We demonstrated the possibility of integrating new mathematical models using the two mathematical models.

*Question 1.1: How can we assess performance of the models in the system?*

Since all models use the same laboratory data, comparing the prediction results against the reference data can provide insights into the accuracy of a certain model for a given disease. As a result, the prediction service incorporated an evaluation module that implements NRMSE.

*Question 2: How can we visualize prediction results in a way that facilitates access to prediction results and support decision making?*

We designed and implemented user interfaces to visualize the spatio-temporal prediction results using map-based interfaces. The system also allows users to interactively upload and display their own geospatial data files in csv format.

In general, the thesis can significantly improve the status of disease prediction systems, investment and time of development. The framework also speeds up mathematical modeling through its integrated environment for testing and evaluating different mathematical models with respect to other models. Thus, the project contributes to improve the overall disease prediction accuracy and increase the benefits from prediction.

## 9.2 Future Work

1. Addition of possible outbreak control strategies evaluation

In this thesis, we did not focus on interactive decision support for evaluating potential disease mitigation strategies (i.e. school closures and vaccination). In the future, a study can be done on the extension of the system to include such decision support capability.

STEM allows assessment of disease prevention, intervention, and response strategies, such as vaccination program, isolating infected individuals, implementing social distancing, evacuation of a region, shutting down air transportation, closing a road or preventing mixing of infected individuals across borders.

The STEM interface we have created in this thesis only perform basic operations, however more advanced interface can be created in order to utilize these and other functionalities from the prediction service.

The Bayesian model does not have such functionalities, but it can be possible to extend a capability for assessment of some disease intervention strategies, such as shutting down air transportation, closing a road across borders.

2. Evaluation of the visualization

Evaluation studies of the visualization are required to find its effect on users' (i.e. GP's and public health officials) decision making. Controlled lab experiments helps to improve the design of visualization systems. But, more studies are required to provide actionable evidence to promote the adoption of the visualization. Even so, there was not enough controlled experiment in this study.

3. Addition of data privacy and access control

When infectious disease datasets are shared across jurisdictions, access control and data privacy issues need to be resolved between the involved parties. The prediction service could be extended to include components that control the exchange of information and ensure healthcare data privacy. However, currently the prediction system only has access to the Snow system, which only provides anonymized and aggregated data.

# Bibliography

A. Rvachev, L. & Longini Jr., I. M. (1985), 'A mathematical model for the global spread of influenza', *Mathematical Biosciences* **75**(1), 3–22.
**URL:** *http://www.sciencedirect.com/science/article/pii/0025556485900641*

Acquah, H. & Carlo, M. (2010), 'Comparison of akaike information criterion (AIC) and bayesian information criterion (BIC) in selection of an asymmetric price relationship', *Agricultural Economics* **2**(1), 001–006.

Aguirre, A. & Gonzalez, E. (1992), 'The feasibility of forecasting influenza epidemics in cuba', *Memórias Do Instituto Oswaldo Cruz* **87**(3), 429–432. PMID: 1343651.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/1343651*

Andreasen, V., Viboud, C. & Simonsen, L. (2008), 'Epidemiologic characterization of the 1918 influenza pandemic summer wave in copenhagen: Implications for pandemic control strategies', *Journal of Infectious Diseases* **197**(2), 270–278.
**URL:** *http://jid.oxfordjournals.org/content/197/2/270*

Bailey, N. T. J. (1975), *The Mathematical Theory of Infectious Diseases and Its Applications*, Griffin.

Bellika, J. G., Hasvold, T. & Hartvigsen, G. (2007), 'Propagation of program control: a tool for distributed disease surveillance', *International Journal of Medical Informatics* **76**(4), 313–329. PMID: 16621681.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/16621681*

Bellika, J. G., Ilebrekke, L., Bakkevoll, P. A., Johansen, H., Scholl, J. & Johansen, M. A. (2009), 'Authentication and encryption in the snow disease surveillance network', *Studies in Health Technology and Informatics* **150**, 725–729. PMID: 19745406.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/19745406*

Besag, J. (1974), 'Spatial interaction and the statistical analysis of lattice systems', *Journal of the Royal Statistical Society. Series B* **36**(2), 192–236.

Bisset, K. R., Chen, J., Feng, X., Kumar, V. S. & Marathe, M. V. (2009), Epifast: a fast algorithm for large scale realistic epidemic simulations on distributed memory systems, *in* 'Proceedings of the 23rd international conference on Supercomputing', pp. 430–439.

Brauer, F. (2009), 'Mathematical epidemiology is not an oxymoron', *BMC Public Health* **9 Suppl 1**, S2. PMID: 19922686.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/19922686*

Broeck, W., Gioannini, C., Gonçalves, B., Quaggiotto, M., Colizza, V. & Vespignani, A. (2011), 'The GLEaMviz computational tool, a publicly available software to explore realistic epidemic spreading scenarios at the global scale', *BMC infectious diseases* **11**(1), 37.

Burstein, F., Holsapple, C., Zhu, B. & Chen, H. (2008), Information visualization for decision support, *in* 'Handbook on Decision Support Systems 2', International Handbooks on Information Systems, Springer Berlin Heidelberg, pp. 699–722.
**URL:** *http://www.springerlink.com/content/q506212191h6413l/abstract/*

Cakici, B. (2010), 'Case: a framework for computer supported outbreak detection.', *BMC medical informatics and decision making* **10**, 14.

CDC (2011*a*), 'Appendix d: The healthmap system'.
**URL:** *http://wwwnc.cdc.gov/travel/yellowbook/2012/appendices/appendix-d-the-healthmap-system.htm*

CDC (2011*b*), 'Community flu 1.0'.
**URL:** *http://www.cdc.gov/flu/tools/communityflu/*

Chang, M., Glynn, M. & Groseclose, S. (2003), 'Endemic, notifiable bioterrorism-related diseases, united states, 1992-1999', *Emerging Infectious Diseases* **9**, 556.

Chao, D. L., Halloran, M. E., Obenchain, V. J. & Longini, Ira M, J. (2010), 'Flute, a publicly available stochastic influenza epidemic simulation model', *PLoS computational biology* **6**(1), e1000656. PMID: 20126529.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/20126529*

Cheng, C. K., Lau, E. H., Ip, D. K., Yeung, A. S., Ho, L. M. & Cowling, B. J. (2009), 'A profile of the online dissemination of national influenza surveillance data', *BMC Public Health* **9**, 339. PMID: 19754978 PMCID: 2754460.

Ching, P., Harriman, K., Li, Y., Pessoa-Silva, C. L., Seto, W. & Wang, T. K. (2007), Infection prevention and control of epidemic- and pandemic-prone acute respiratory diseases in health care, Technical report, WHO.
**URL:** *http://www.who.int/csr/resources/publications/WHO_CDS_EPR_2007_6c.pdf*

Clarke, D. & Smith, S. (2008), 'Introducing EclipseLink'.
**URL:** *http://eclipse.dzone.com/articles/introducing-eclipselink?page=0,3*

Coiera, E. (2003), *Guide to Health Informatics*, 2 edn, Hodder Arnold Publishers.

Connell, R., Dawson, P. & Skvortsov, A. (2009), Comparison of an agent-based model of disease propagation with the generalised SIR epidemic model, Technical report, DTIC Document.

Dawood, F. S., Jain, S., Finelli, L., Shaw, M. W., Lindstrom, S., Garten, R. J., Gubareva, L. V., Xu, X., Bridges, C. B. & Uyeki, T. M. (2009), 'Emergence of a novel swine-origin influenza a (H1N1) virus in humans', *The New England Journal of Medicine* **360**(25), 2605–2615. PMID: 19423869.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/19423869*

Denning, P. J., Comer, D. E., Gries, D., Mulder, M. C., Tucker, A., Turner, A. J. & Young, P. R. (1989), 'Computing as a discipline', *Commun. ACM* **32**(1), 9–23.
**URL:** *http://doi.acm.org/10.1145/63238.63239*

Dull, R. B. & Tegarden, D. P. (1999), 'A comparison of three visual representations of complex multidimensional accounting information', *Journal of Information Systems* p. 117–131.

Earnest, A., Tan, S. B., Wilder-Smith, A. & Machin, D. (2012), 'Comparing statistical models to predict dengue fever notifications', *Computational and Mathematical Methods in Medicine* **2012**, 1–6.
**URL:** *http://www.hindawi.com/journals/cmmm/2012/758674/*

Edlund, S. B., Davis, M. A. & Kaufman, J. H. (2010), The spatiotemporal epidemiological modeler, *in* 'Proceedings of the 1st ACM International Health Informatics

Symposium', IHI '10, ACM, New York, NY, USA, p. 817–820.
**URL:** *http://doi.acm.org/10.1145/1882992.1883115*

Edlund, S., Bromberg, M., Chodick, G., Douglas, J., Ford, D., Kaufman, Z., Lessler, J., Marom, R., Mesika, Y., Ram, R., Shalev, V. & Kaufman, J. (2011a), 'A spatiotemporal model for seasonal influenza', *electronic Journal of Health Informatics* **6**(1), e9.
**URL:** *http://www.ejhi.net/ojs/index.php/ejhi/article/view/123/87*

Edlund, S., Kaufman, J., Lessler, J., Douglas, J., Bromberg, M., Kaufman, Z., Bassal, R., Chodick, G., Marom, R., Shalev, V., Mesika, Y., Ram, R. & Leventhal, A. (2011b), 'Comparing three basic models for seasonal influenza', *Epidemics* **3**(3), 135–142.
**URL:** *http://www.sciencedirect.com/science/article/pii/S1755436511000259*

Eichner, M., Schwehm, M., Duerr, H. P. & Brockmann, S. O. (2007), 'The influenza pandemic preparedness planning tool InfluSim', *BMC infectious diseases* **7**(1), 17.

Evaluation, M. (2009), Making research findings actionable, Technical report, MEASURE Evaluation.
**URL:** *http://www.k4health.org/system/files/Making Research Findings Actionable.pdf*

Fong, I. & Alibek, K., eds (2009), *Bioterrorism and Infectious Agents: A New Dilemma for the 21st Century*, 1 edn, Springer.

Ford, D., Kaufman, J. & Eiron, I. (2006), 'An extensible spatial and temporal epidemiological modelling system', *International Journal of Health Geographics* **5**(1), 4.

Foundation, A. S. (2010), 'Xerces2 java parser readme'.
**URL:** *http://xerces.apache.org/xerces2-j/*

Freifeld, C. C., Mandl, K. D., Reis, B. Y. & Brownstein, J. S. (2008), 'HealthMap: global infectious disease monitoring through automated classification and visualization of internet media reports', *Journal of the American Medical Informatics Association* **15**(2), 150–157.
**URL:** *http://jamia.bmj.com/content/15/2/150*

Frerichs, R. R. (2006), 'John snow', http://www.ph.ucla.edu/epi/snow.html.
**URL:** *http://www.ph.ucla.edu/epi/snow.html*

Gani, J. (1980), 'Mathematical models of epidemics', *The Mathematical Intelligencer* **3**(1), 41–43.
**URL:** *http://www.springerlink.com/content/hv3k5x3750n56rqg/*

Gao, S., Mioc, D., Anton, F., Yi, X. & Coleman, D. J. (2008), 'Online GIS services for mapping and sharing disease information', *International Journal of Health Geographics* **7**(1), 8.
**URL:** *http://www.ij-healthgeographics.com/content/7/1/8*

Geilhufe et al. (2012), Spatio-temporal modeling of communicable diseases: A case study of north norway. Unpublished manuscript.

Gonzalez, A. (2010), 'Java,xml and xstream'.

Grassly, N. C. & Fraser, C. (2008), 'Mathematical models of infectious disease transmission', *Nat Rev Micro* **6**(6), 477–487.
**URL:** *http://dx.doi.org/10.1038/nrmicro1845*

Green, M. S., Swartz, T., Mayshar, E., Lev, B., Leventhal, A., Slater, P. E. & Shemer, J. (2002), 'When is an epidemic an epidemic?', *The Israel Medical Association Journal: IMAJ* **4**(1), 3–6. PMID: 11802306.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/11802306*

Guzman Herrador, B., Aavitsland, P., Feiring, B., Riise Bergsaker, M. & Borgen, K. (2012), 'Usefulness of health registries when estimating vaccine effectiveness during the influenza A(H1N1)pdm09 pandemic in norway', *BMC Infectious Diseases* **12**(1), 63.
**URL:** *http://www.biomedcentral.com/1471-2334/12/63*

Han, X., Vlas, D., J, S., Fang, L.-Q., Feng, D., Cao, W. & Habbema, J. D. F. (2009), 'Mathematical modelling of SARS and other infectious diseases in china: a review', *Tropical Medicine & International Health* **14**(s1), 92–100.
**URL:** *http://onlinelibrary.wiley.com/doi/10.1111/j.1365-3156.2009.02244.x/abstract*

Hartvigsen, G. & Pedersen, S. (2012), Lessons learned from 20 years with telemedicine in north norway.

Heisterkamp, S. H., Dekkers, A. L. M. & Heijne, J. C. M. (2006), 'Automated detection of infectious disease outbreaks: hierarchical time series models', *Statistics in*

*Medicine* **25**(24), 4179–4196. PMID: 16958149.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/16958149*

High Jr, R., Kinder, S. & Graham, S. (2005), 'IBM's SOA foundation', *An Architectural Introduction and Overview. Whitepaper IBM* p. 68.
**URL:** *http://public.dhe.ibm.com/software/dw/webservices/ws-soa-whitepaper.pdf*

Holand, A., Steinsland, I., Martino, S. & Jensen, H. (2011), 'Animal models and integrated nested laplace approximations', p. 30.
**URL:** *http://www.math.ntnu.no/preprint/statistics/2011/S4-2011.pdf*

Hufnagel, L., Brockmann, D. & Geisel, T. (2004), 'Forecast and control of epidemics in a globalized world', *Proceedings of the National Academy of Sciences of the United States of America* **101**(42), 15124–15129. PMID: 15477600.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/15477600*

IBM (2007), 'IBM contributes software that predicts spread of emerging infectious diseases - united states'.
**URL:** *http://www-03.ibm.com/press/us/en/pressrelease/21656.wss*

Jersey (2008), 'Jersey', http://jersey.java.net/.
**URL:** *http://jersey.java.net/*

Kadane, J. & Lazar, N. (2004), 'Methods and criteria for model selection', *Journal of the American Statistical Association* **99**(465), 279–290.

Kaufman, J. (2011), 'Introduction to compartment models'.
**URL:** *http://wiki.eclipse.org/Introduction_to_Compartment_Models*

Kaufman, J., Conant, J. L., Ford, D. A., Kirihata, W., Jones, B. & Douglas, J. V. (2008), Assessing the accuracy of spatiotemporal epidemiological models, *in* D. Zeng, H. Chen, H. Rolka & B. Lober, eds, 'Biosurveillance and Biosecurity', Vol. 5354, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 143–154.
**URL:** *http://www.springerlink.com/content/n5177m15r70q17g7/*

Kaufman, J., Edlund, S. & Douglas, J. (2009), 'Infectious disease modeling: Creating a community to respond to biological threats', *Statistical Communications in Infectious Diseases* **1**(1).
**URL:** *http://www.bepress.com/scid/vol1/iss1/art1*

Keeling, M. & Rohani, P. (2008), *Modeling infectious diseases in humans and animals*, Princeton Univ Pr.

Kellen, V. (2005), 'Decision making and information visualization: Research directions'.
**URL:** *http://www.kellen.net/Visualization_Decision_Making.htm*

Lascano, J. (2008), 'JPA implementations versus pure JDBC'.
**URL:** *http://www.espe.edu.ec/portal/files/sitiocongreso/congreso/c_computacion/PaperJPAversusJDBC_edisonlascano.pdf*

Leer et al. (2012), Disease query engine: Incident based weighting for symptom to disease matching. Unpublished manuscript.

Lessler, J., Kaufman, J. H., Ford, D. A. & Douglas, J. V. (2009), 'The cost of simplifying air travel when modeling disease spread', *PLoS ONE* **4**(2), e4403.
**URL:** *http://dx.doi.org/10.1371/journal.pone.0004403*

Li, T., Feng, S. & Xia Li, L. (2001), 'Information visualization for intelligent decision support systems', *Knowledge-Based Systems* **14**(5–6), 259–262.
**URL:** *http://www.sciencedirect.com/science/article/pii/S0950705101001046*

Lober, W. B., Trigg, L. & Karras, B. (2004), 'Information system architectures for syndromic surveillance', *MMWR Morb Mortal Wkly Rep* **53**, 203–208.
**URL:** *http://www.cdc.gov/Mmwr/preview/mmwrhtml/su5301a37.htm*

Lombardo, J., Burkom, H., Elbert, E., Magruder, S., Lewis, S., Loschen, W., Sari, J., Sniegoski, C., Wojcik, R. & Pavlin, J. (2003), 'A systems overview of the electronic surveillance system for the early notification of community-based epidemics (ESSENCE II)', *Journal of urban health: bulletin of the New York Academy of Medicine* **80**(Supplement 1), i32–i42.

Lowen, A. C., Mubareka, S., Steel, J. & Palese, P. (2007), 'Influenza virus transmission is dependent on relative humidity and temperature', *PLoS Pathog* **3**(10), e151.
**URL:** *http://dx.plos.org/10.1371/journal.ppat.0030151*

Manitz, J. (2010), Automated Detection of Infectious Disease Outbreaks, PhD thesis, Ludwig-Maximilians-Universität München, Germany.
**URL:** *http://epub.ub.uni-muenchen.de/11908/1/DA_Manitz.pdf*

Martino, S. & Rue, H. (2011), 'Case studies in bayesian computation using INLA', *NTNU* p. 16.
**URL:** *http://www.math.ntnu.no/ hrue/r-inla.org/papers/martino-rue-book.pdf*

Massad, E., Burattini, M. N., Lopez, L. F. & Coutinho, F. A. B. (2005), 'Forecasting versus projection models in epidemiology: the case of the SARS epidemics', *Medical Hypotheses* **65**(1), 17–22. PMID: 15893110.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/15893110*

McBryde, E. S. (2006), 'Mathematical and statistical modelling of infectious diseases in hospitals'.
**URL:** *http://eprints.qut.edu.au/16330/*

Meyers, L. (2007), 'Contact network epidemiology: Bond percolation applied to infectious disease prediction and control', *Bulletin-American Mathematical Society* **44**(1), 63.

Miller, G. & Williams, L. (2006), 'Personas: Moving beyond Role-Based requirements engineering', *Microsoft and North Carolina State University* .

Mniszewski, S. M., Del Valle, S. Y., Stroud, P. D., Riese, J. M. & Sydoriak, S. J. (2008), EpiSimS simulation of a multi-component strategy for pandemic influenza, *in* 'Proceedings of the 2008 Spring simulation multiconference', SpringSim '08, Society for Computer Simulation International, San Diego, CA, USA, p. 556–563.
**URL:** *http://dl.acm.org/citation.cfm?id=1400549.1400636*

Myers, M. F., Rogers, D. J., Cox, J., Flahault, A. & Hay, S. I. (2000), 'Forecasting disease risk for increased epidemic preparedness in public health', *Advances in Parasitology* **47**, 309–330. PMID: 10997211.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/10997211*

Nelder, J. A. & Mead, R. (1965), 'A simplex method for function minimization', *The Computer Journal* **7**(4), 308–313.
**URL:** *http://comjnl.oxfordjournals.org/content/7/4/308*

ORACLE (2012), The java EE 6 tutorial.
**URL:** *http://docs.oracle.com/javaee/6/tutorial/doc/javaeetutorial6.pdf*

Pautasso, C., Zimmermann, O. & Leymann, F. (2008), Restful web services vs. big'web services: making the right architectural decision, *in* 'Proceeding of the 17th international conference on World Wide Web', p. 805–814.

Ramirez, L. L. R. (2008), On the dynamics of infectious diseases in non-homogeneous populations, Doctor of philosophy, University of Waterloo.
**URL:** *http://hdl.handle.net/10012/4054*

Reinhardt, M., Elias, J., Albert, J., Frosch, M., Harmsen, D. & Vogel, U. (2008), 'EpiScanGIS: an online geographic surveillance system for meningococcal disease', *International Journal of Health Geographics* **7**(1), 33.
**URL:** *http://www.ij-healthgeographics.com/content/7/1/33*

Reis, B. Y., Kirby, C., Hadden, L. E., Olson, K., McMurry, A. J., Daniel, J. B. & Mandl, K. D. (2007), 'AEGIS: a robust and scalable real-time public health surveillance system', *Journal of the American Medical Informatics Association* **14**(5), 581 –588.
**URL:** *http://jamia.bmj.com/content/14/5/581.abstract*

Robertson, S. & Robertson, J. (1999), *Mastering the Requirements Process*, Addison-Wesley Professional.

Rue, H. & Held, L. (2005), *Gaussian Markov Random Fields: Theory and Applications*, Chapman & Hall, London, UK.

Rue, H., Martino, S. & Chopin, N. (2009), 'Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations', *Journal Of The Royal Statistical Society Series B* **71**(2), 319–392.
**URL:** *http://ideas.repec.org/a/bla/jorssb/v71y2009i2p319-392.html*

Sattenspiel, L. (1990), 'Modeling the spread of infectious disease in human populations', *American Journal of Physical Anthropology* **33**(S11), 245–276.
**URL:** *http://onlinelibrary.wiley.com/doi/10.1002/ajpa.1330330511/abstract*

Schrödle, B. & Held, L. (2011), 'Spatio-temporal disease mapping using INLA', *Environmetrics* **22**(6), 725–734.
**URL:** *http://onlinelibrary.wiley.com/doi/10.1002/env.1065/abstract*

Schrödle, B., Held, L., Riebler, A. & Danuser, J. (2011), 'Using integrated nested laplace approximations for the evaluation of veterinary surveillance data from

switzerland: a case study', *Journal of the Royal Statistical Society: Series C (Applied Statistics)* **60**(2), 261–279.
**URL:** *http://onlinelibrary.wiley.com/doi/10.1111/j.1467-9876.2010.00740.x/abstract*

Schunn, C. D. & Wallach, S. B. (2005), Evaluating goodness-of-fit in comparison of models to data, *in* 'Psychologie der Kognition: Reden und Vorträge anlässlich der Emeritierung von Werner H. Tack', University of Saarland Press, Germany.

Shaman, J. & Kohn, M. (2009), 'Absolute humidity modulates influenza survival, transmission, and seasonality', *Proceedings of the National Academy of Sciences* .
**URL:** *http://www.pnas.org/content/early/2009/02/09/0806852106*

Shaman, J., Pitzer, V. E., Viboud, C., Grenfell, B. T. & Lipsitch, M. (2010), 'Absolute humidity and the seasonal onset of influenza in the continental united states', *PLoS Biol* **8**(2), e1000316.
**URL:** *http://dx.doi.org/10.1371/journal.pbio.1000316*

Skrøvseth, S. O., Bellika, J. G. & Godtliebsen, F. (2012), Causality in scale space as an approach to change detection. Unpublished article.

Spencer, S. (2008), 'Stochastic epidemic models for emerging diseases'.

Statistics Norway (2002), 'Population and area in urban settlements in 1 january 2002'.
**URL:** *http://www.ssb.no/english/subjects/02/01/10/beftett_en/arkiv/tab-2002-09-03-03-en.html*

Statistics Norway (2011), 'Quarterly population changes'.
**URL:** *http://www.ssb.no/folkendrkv_en/tab-2011-11-17-01-en.html*

Tsui, F., Espino, J. U., Dato, V. M., Gesteland, P. H., Hutman, J. & Wagner, M. M. (2003), 'Technical description of RODS: a real-time public health surveillance system', *Journal of the American Medical Informatics Association: JAMIA* **10**(5), 399–408. PMID: 12807803.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/12807803*

Venables, W. N. & Smith, D. M. (2012), 'An introduction to r'.
**URL:** *http://cran.r-project.org/doc/manuals/R-intro.pdf*

Vessey, I. & Galletta, D. (1991), 'Cognitive fit: An empirical study of information acquisition', *Information Systems Research* **2**(1), 63–84.
**URL:** *http://isr.journal.informs.org/content/2/1/63*

Villa, E. d., McPherson, M. & Ramirez, L. L. R. (2011), 'Simid: Simulation of infectious disease'.

White, D. E. & Fenner, F. J. (1994), *Medical Virology, Fourth Edition*, 4 edn, Academic Press.

WHO (2005), 'International health regulation'.
**URL:** *http://whqlibdoc.who.int/publications/2008/9789241580410_eng.pdf*

WHO (2012), 'Disease outbreaks'.
**URL:** *http://www.who.int/topics/disease_outbreaks/en/*

Willgert, K. J. E., Schroedle, B. & Schwermer, H. (2011), 'Spatial analysis of bluetongue cases and vaccination of swiss cattle in 2008 and 2009', *Geospatial Health* **5**(2), 227–237. PMID: 21590673.
**URL:** *http://www.ncbi.nlm.nih.gov/pubmed/21590673*

Yi, Q., Hoskins, R. E., Hillringhouse, E. A., Sorensen, S. S., Oberle, M. W., Fuller, S. S. & Wallace, J. C. (2008), 'Integrating open-source technologies to build low-cost information systems for improved access to public health data', *International Journal of Health Geographics* **7**(1), 29.
**URL:** *http://www.ij-healthgeographics.com/content/7/1/29/abstract*

Zeng, D., Chen, H., Castillo-Chavez, C., Lober, W. & Thurmond, M. (2011), *Infectious Disease Informatics and Biosurveillance*, Integrated Series in Information Systems, 1st edn.
**URL:** *http://www.springer.com/public+health/book/978-1-4419-6891-3*

Zhang, P. (2006), *Human-Computer Interaction and Management Information Systems: Foundations*, M.E. Sharpe.

# Appendix A

# Prediction result xml schema definition

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="prediction" type="predictionType">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="area" minOccurs="0" maxOccurs="unbounded" use="required"/>
          <xs:complexType>
              <xs:sequence >
                <xs:element name="result" type="xs:double" use="required"/>
              </xs:sequence>
              <xs:attribute name="code" type="xs:int" use="required"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="period" type="xs:int" use="required"/>
      <xs:attribute name="date" type="xs:date" use="required"/>
      <xs:attribute name="model" type="xs:string" use="required"/>
    </xs:complexType>
</xs:element>
</xs:schema>
```

Figure A.1: Prediction result xml schema definition

# Appendix B

# Prediction schedule xml schema definition

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="schedules" type="predictionType">
    <xs:complexType>
       <xs:sequence>
            <xs:element name="schedule" maxOccurs="unbounded" use="required"/>
                <xs:complexType>
                    <xs:sequence >
                       <xs:element name="id" type="xs:int" use="required"/>
                       <xs:element name="infectious_agent" type="xs:string" use="required"/>
                       <xs:element name="model" type="xs:string" use="required"/>
                       <xs:element name="date" type="xs:dateTime" use="required"/>
                       <xs:element name="period" type="xs:int" use="required"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
       </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:schema>
```

Figure B.1: Prediction schedule xml schema definition

# Appendix C

# Municipalities of Troms and Finnmark counties

Table C.1: Municipalities of Troms county and ISO-code

| Municipality | ISO-code |
|---|---|
| Harstad | NO-1901 |
| Tromsø | NO-1902 |
| Kvaefjord | NO-1911 |
| Skaanland | NO-1913 |
| Bjarkoy | NO-1915 |
| Ibestad | NO-1917 |
| Gratangen | NO-1919 |
| Lavangen | NO-1920 |
| Bardu | NO-1922 |
| Salangen | NO-1923 |
| Maalselv | NO-1924 |
| Sorreisa | NO-1925 |
| Dyroy | NO-1926 |
| Tranoy | NO-1927 |
| Torsken | NO-1928 |
| Berg | NO-1929 |
| Lenvik | NO-1931 |
| Balsfjord | NO-1933 |
| Karlsoy | NO-1936 |
| Lyngen | NO-1938 |
| Storfjord | NO-1939 |
| Kaafjord | NO-1940 |
| Skjervoy | NO-1941 |
| Nordreisa | NO-1942 |
| Kvaenangen | NO-1943 |

Table C.2: Municipalities of Finnmark county and ISO-code

| Municipality | ISO-code |
|---|---|
| Vardo | NO-2002 |
| Vadso | NO-2003 |
| Hammerfest | NO-2004 |
| Kautokeino | NO-2011 |
| Alta | NO-2012 |
| Loppa | NO-2014 |
| Hasvik | NO-2015 |
| Kvalsund | NO-2017 |
| Maasoy | NO-2018 |
| Nordkapp | NO-2019 |
| Porsanger | NO-2020 |
| Karasjok | NO-2021 |
| Lebesby | NO-2022 |
| Gamvik | NO-2023 |
| Berlevaag | NO-2024 |
| Tana | NO-2025 |
| Nesseby | NO-2027 |
| Baatsfjord | NO-2028 |
| Sor-Varanger | NO-2030 |

# Appendix D

# Bayesian Model Weekly and Monthly Predictions

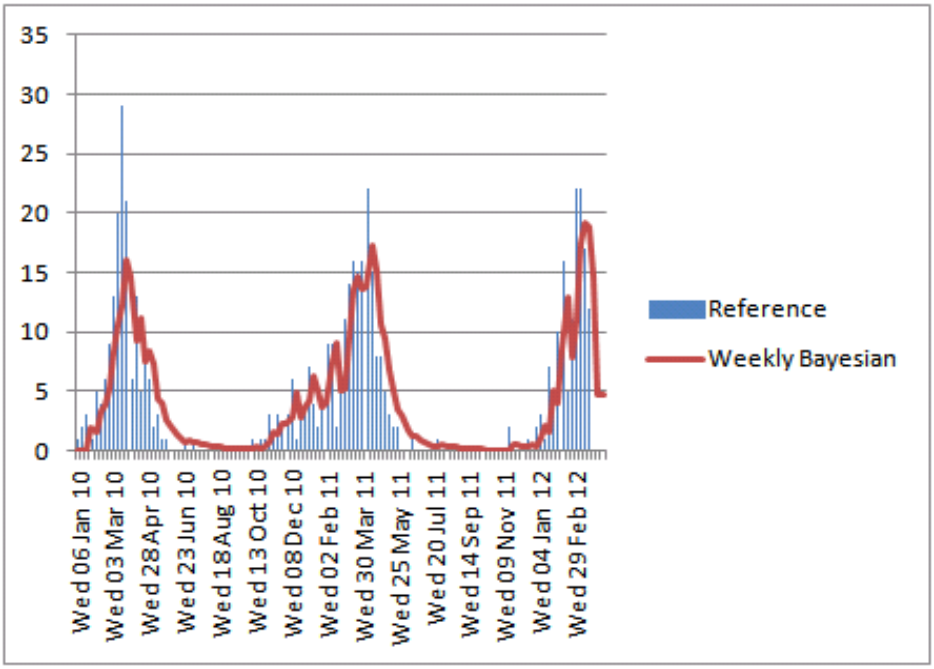



Figure D.1: Weekly RS-virus cases (summed over all locations) for actual RS-virus (blue) and Bayesian model (red)

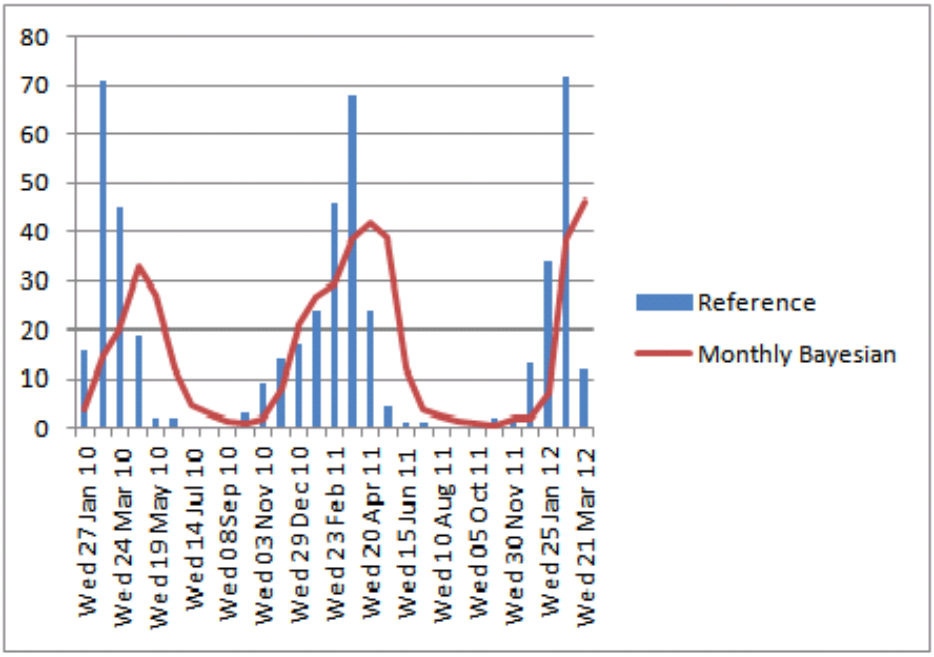



Figure D.2: Monthly RS-virus cases (summed over all locations) for actual RS-virus (blue) and Bayesian model (red)

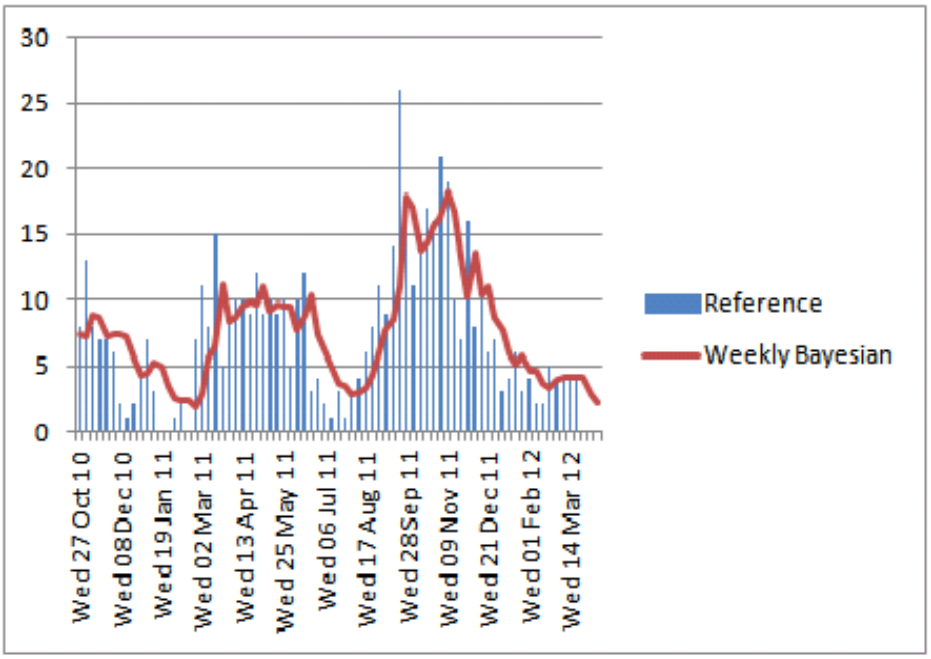



Figure D.3: Weekly Rhinovirus cases (summed over all locations) for actual Rhinovirus (blue) and Bayesian model (red)

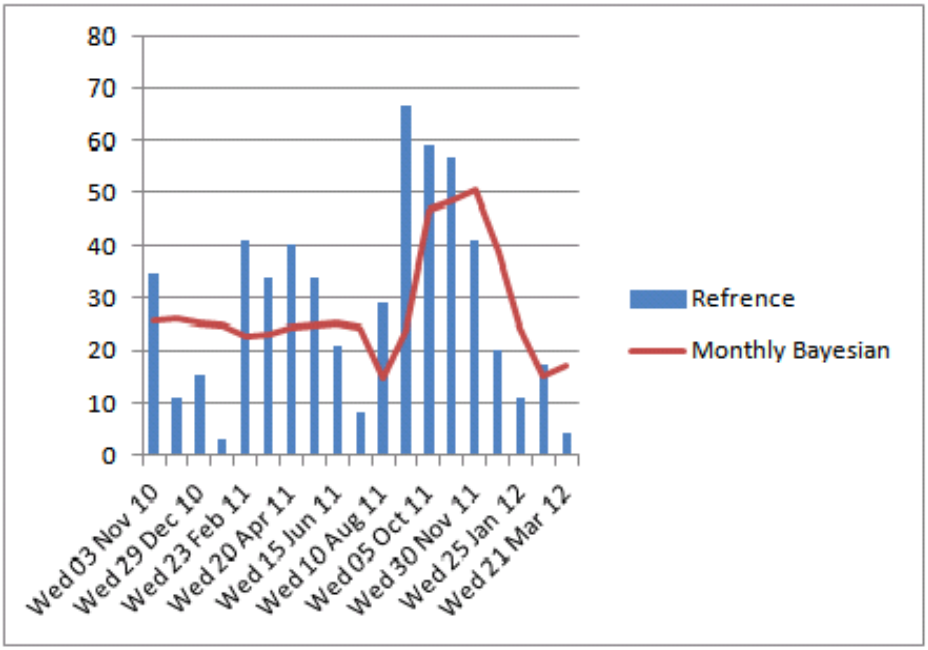



Figure D.4: Monthly Rhinovirus cases (summed over all locations) for actual Rhinovirus (blue) and Bayesian model (red)

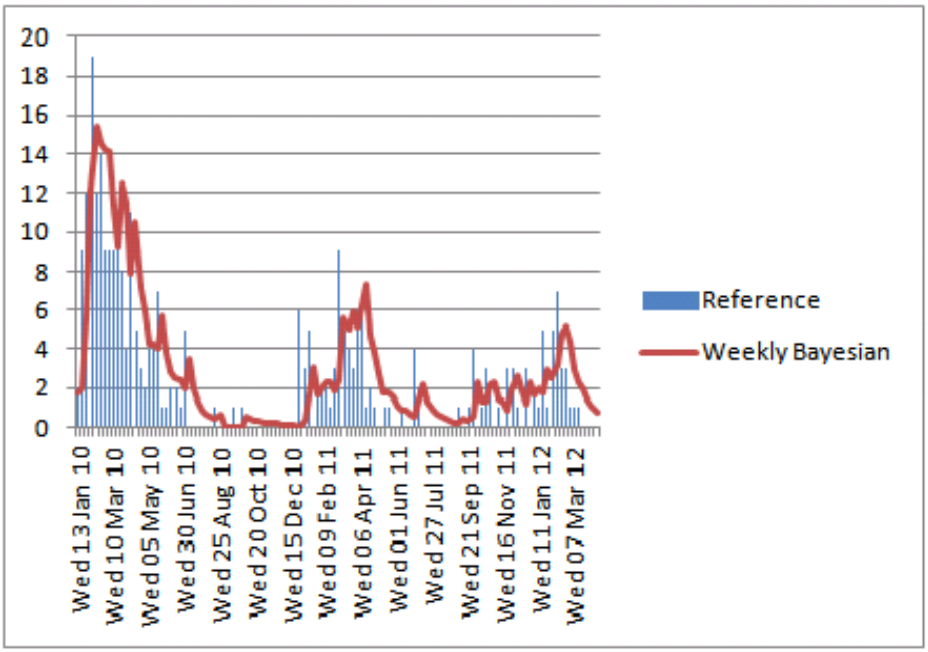



Figure D.5: Weekly Norovirus cases (summed over all locations) for actual Norovirus (blue) and Bayesian model (red)

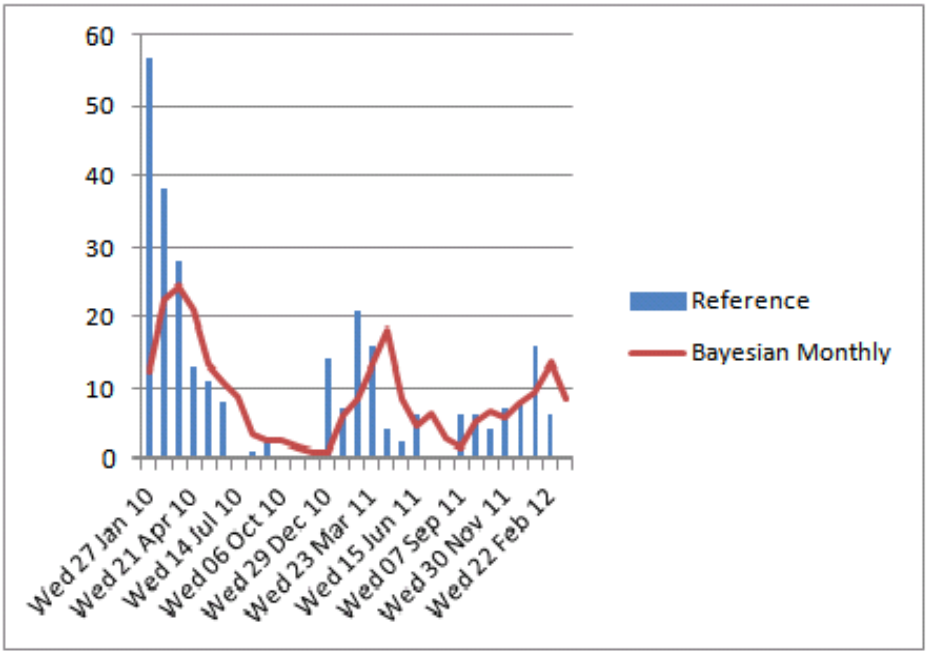



Figure D.6: Monthly Norovirus cases (summed over all locations) for actual Norovirus (blue) and Bayesian model (red)

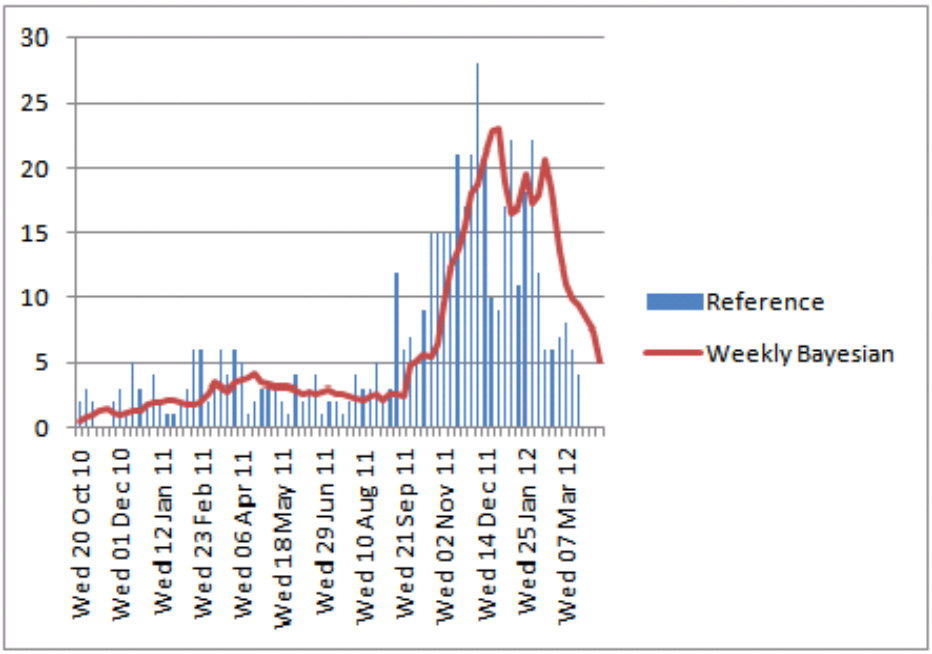



Figure D.7: Weekly Mycoplasma pneumoniae cases (summed over all locations) for actual Mycoplasma pneumoniae (blue) and Bayesian model (red)

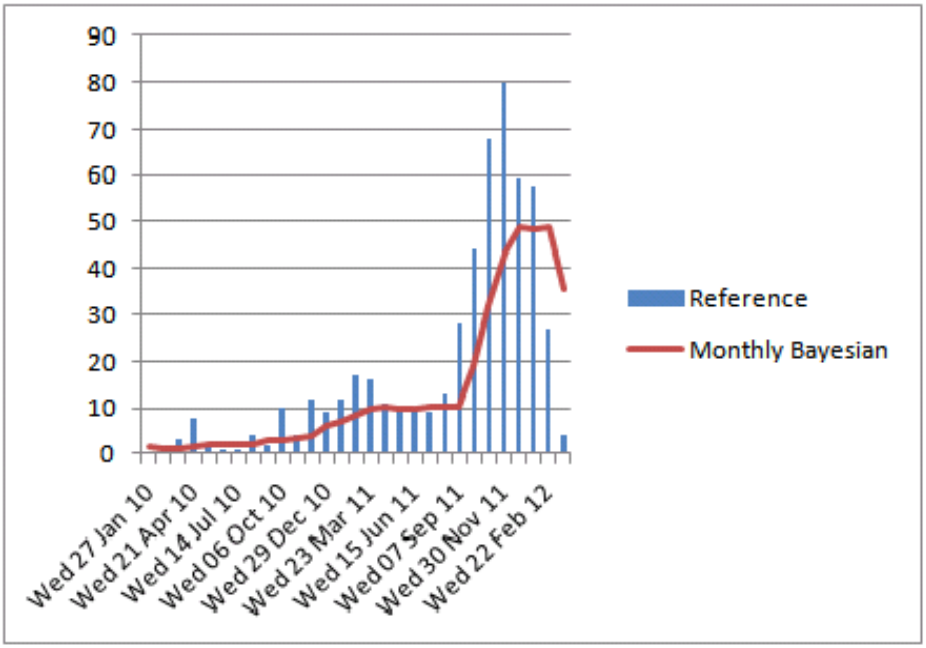



Figure D.8: Monthly Mycoplasma pneumoniae cases (summed over all locations) for actual Mycoplasma pneumoniae (blue) and Bayesian model (red)