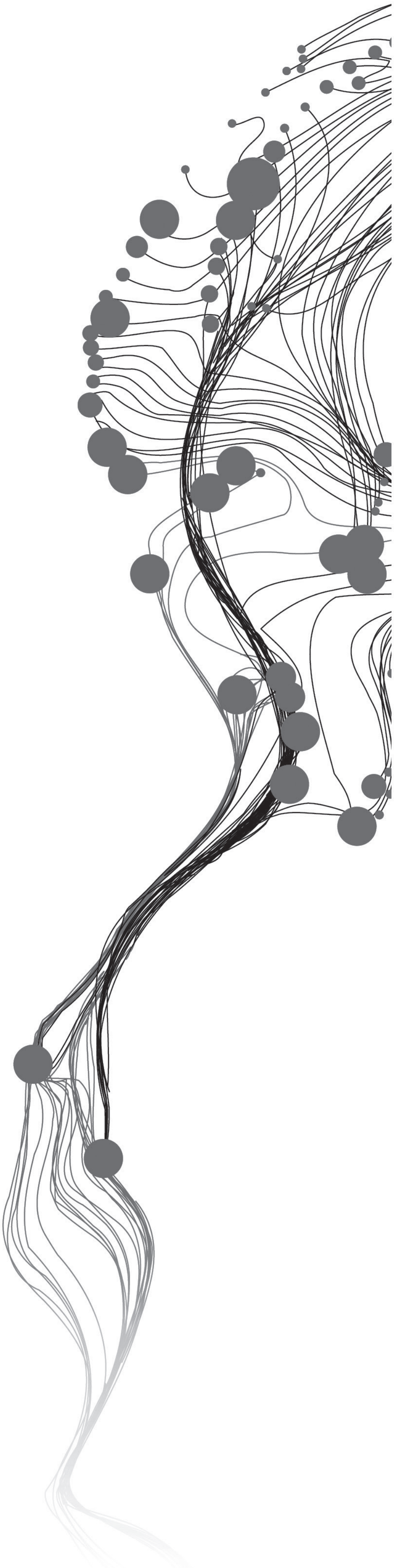
FACILITATING REAL-TIME INUNDATION PREDICTION USING GEO WEB SERVICES

KINGSLEY NNAEMEKA OBI
February, 2012

SUPERVISORS:

Dr. U.D. Turdukulov

Dr. Ir. R.L.G Lemmens
Dr. M.W. Straatsma



FACILITATING REAL-TIME INUNDATION PREDICTION USING GEO WEB SERVICES

KINGSLEY NNAEMEKA OBI
Enschede, The Netherlands, February, 2012

Thesis submitted to the Faculty of Geo-information Science and Earth Observation of the University of Twente in partial fulfilment of the requirements for the degree of Master of Science in Geo-information Science and Earth Observation.
Specialization: Geoinformatics

SUPERVISORS:

Dr. U.D. Turdukulov

Dr. Ir. R.L.G Lemmens
Dr. M.W. Straatsma

THESIS ASSESSMENT BOARD:

Dr. Ir. R. A. de By (chair)

Ir. J. Ticheler (GeoCat B.V.)

ABSTRACT

Real time inundation prediction is a challenge in many areas in the world which lack sufficient data. Such challenging scenarios are the July 2010 Pakistan flood, the May 2011 Colombian flood and the October 2011 Thailand floods. In these cases, future flood extents could not be predicted from the current flood extents and timely information could not passed in a consistent manner. Thereby leading to losses of lives and properties. Effective prediction of real time inundation is a very time and data consuming process for flood modelers. The recent development in web technology, standards and protocols could make predictions faster in these data scarce regions. This research involves exploring the use of web service for inundation prediction modeling since nowadays large amounts of geospatial data (raster and vector) are available on the internet as web services. Presently, there are two major web standard protocols for sharing remote sensing and gridded data. The Open-source Project for a Network Data Access Protocol (OPeNDAP) used largely by the Earth Science (ES) communities and the Open Geospatial Consortium's (OGC) Web Coverage Service (WCS) standard used mainly by the GIS communities. For the discovery of these data, both ES and GIS communities adopts different catalogue standards. ES community use the Thematic Realtime Environmental Distributed Data Services (THREDDS) as catalogue service for the discovery of data. Similarly, the OGC's Catalogue Service for Web (CS/W) designed to support the registry and discovery of geospatial information is used by the GIS communities. There is a need to use data discovered from both catalogues for timely inundation prediction modeling. But there are challenges in interoperable access of metadata between both communities. Two experiments were conducted. The first, was done by implementing a unified data discovery mechanism by harvesting THREDDS dataset records into GeoNetwork CS/W. Thereby creating a rich metadata for datasets used for inundation prediction modeling from both the ES and GIS communities. The experiment also focused on the problems faced in harvesting THREDDS records into GeoNetwork and discussed solutions to such problems. The second experiment involved retrieving dataset after they have been found from the data discovery mechanism. We designed a three-tier stand-alone prototype web service which retrieved datasets based on the access URL and bounding box given by the user. The experiment accessed both OPeNDAP and WCS servers using open source python libraries Pydap and OWSLib respectively as clients to these servers. Both experiments formed an integral part of a loosely coupled prototype web service infrastructure, that could facilitate the consistent discovery and access to datasets required as inputs in hydrodynamic models for inundation prediction modeling. The datasets gotten the web service infrastructure was visualized using client software, so users could access its applicability based on the quality of datasets for inundation prediction modeling. The infrastructure was partially evaluated, with each component evaluated as a separate unit using certain criteria like spatial and temporal subsetting. But this infrastructure equally had its limitations.

In all, these experiments served as a proof of concept demonstration that would enable flood modelers (users) further explore the accuracy and suitability of data gotten from these web services for inundation prediction modeling.

Keywords

WCS, OPeNDAP, THREDDS, CS/W, Web services (geo web services), Inundation prediction, hydrodynamic model, inundation prediction modeling (flood extent modeling)

ACKNOWLEDGEMENTS

First, I want to thank God Almighty for the grace He has given me for making my program a huge success.

I want to acknowledge the scholarship from the government of the Netherlands under Netherlands Fellowship Program (NFP) and University of Twente, Faculty of Geo-Information Science and Earth Observation (ITC) for granting me admission, awarding me the scholarship and imbibing me with the academic skills required to develop my country Nigeria.

I am greatly indebted to my first supervisor Dr. Ulanbek Turdukulov. For his profound knowledge, guidance, encouragement and whose valuable insight played a critical role in the shape of this research. I would also like to express my gratitude to my second supervisor Dr. Rob Lemmens, whose experience, suggestions and feedback improved the significance of my research. In addition I would like show my appreciation to my third supervisor Dr. Menno Straatsma, whose suggestions and timely feedback defined the scope and significance of my research. I want to express my thanks to Mr. Bas Retsios, for his patience, immense technical expertise and support.

In addition, I would like to thank all my colleagues, the Master of Science graduating set of 2012, Geoinformatics Department, UT/ITC, I found a family in them.

I would like to thank my parents Mr. and Mrs. L. A. Obi for their support and advice in my studies and instructions as parents. Finally, I would like to show my sincere and deepest appreciation to my best friend Chinazor Okafor for her advice, financial support, motivation and encouragement throughout the entire course of my programme.

TABLE OF CONTENTS

Abstract **i**

Acknowledgements **ii**

1 Introduction **1**
1.1 Motivation and problem statement . . . 1
1.2 Research identification . . . 2
1.2.1 Research objective . . . 2
1.2.2 Research sub-objectives . . . 3
1.2.3 Research questions . . . 3
1.2.4 Research hypothesis . . . 3
1.2.5 Innovation aimed at . . . 4
1.2.6 Related work . . . 4
1.3 Project set-up . . . 4
1.3.1 Method adopted . . . 4
1.4 Organization of thesis . . . 5

2 Literature review and user requirements **7**
2.1 Users and users requirements for real-time inundation prediction modeling . . . 7
2.1.1 Users requirements on data used for inundation prediction modeling and their characteristics . . . 7
2.2 Web services . . . 9
2.3 OGC web services . . . 9
2.3.1 Web Coverage Service (WCS) . . . 10
2.3.2 Web Feature Service (WFS) . . . 11
2.3.3 Catalogue Service for the Web (CS/W) . . . 12
2.4 OPeNDAP . . . 12
2.5 THREDDS . . . 14
2.6 Previous work on data provision through web services . . . 15
2.7 Overview of WCS, OPeNDAP, CS/W and THREDDS servers having flood related data . . . 17
2.8 Conclusion . . . 17

3 Design and implementation for common data discovery mechanism **19**
3.1 Interoperability issues between THREDDS and OGC CS/W . . . 19
3.1.1 Metadata schema for THREDDS catalogue . . . 19
3.1.2 Metadata schema for CS/W . . . 21
3.2 Conceptual framework for data discovery . . . 21
3.3 Implementation of data discovery mechanism . . . 25
3.4 Challenges for unified data discovery . . . 26
3.4.1 Structure mapping . . . 28
3.4.2 Syntax mapping . . . 29
3.5 Conclusion . . . 29

4 Design and implementation of data acquisition mechanism 31
4.1 Conceptual framework for data acquisition mechanism 31
4.2 Implementation of data acquisition mechanism 31
4.2.1 The client-side 31
4.2.2 Middleware for data acquisition mechanism 31
4.3 Challenges in implementation of data acquisition mechanism 35
4.4 Conclusion 35

5 Design of web service infrastructure 37
5.1 Conceptual framework for web service infrastructure 37
5.2 Design of software stack for web service infrastructure 37
5.2.1 Choice of clients 37
5.2.2 Client - side user interface 40
5.2.3 Middleware for web service infrastructure 40
5.2.4 Data Servers for web service infrastructure 41
5.3 Evaluation of web service infrastructure 41
5.3.1 Visualization of data residing on the web service infrastructure 42
5.3.2 Applicability of the web service infrastructure for users (flood modelers) 45
5.4 Conclusion 46

6 Results, Discussion, Conclusion and Recommendations 47
6.1 Results 47
6.2 Discussion 48
6.3 Conclusion 49
6.4 Recommendations for further research 51

A Implementation XML file 57

LIST OF FIGURES

1.1 Research framework to show method adopted 6

2.1 User requirements for hydrodynamic modeling 9
2.2 Service-Oriented Architecture (SOA) diagram 10
2.3 Basic CS/W operations 13

3.1 Metadata basic classes 22
3.2 THREDDS ingested into CS/W server with CS/W enabled client 23
3.3 CS/W ingested into THREDDS server with custom built client 24
3.4 CS/W ingested into THREDDS server with THREDDS enabled client 25
3.5 Figure of the HarvestThredds application 27
3.6 Converted metadata of modis product viewed on GeoNetwork interface 27
3.7 Converted metadata of topography viewed on GeoNetwork interface 28

4.1 Sequence diagram for data acquisition 32
4.2 User interface for data acquisition mechanism 32
4.3 Mechansim for data acquisition 33

5.1 Conceptual framework for web service infrastructure 38
5.2 Sequence diagram for web service infrastructure 39
5.3 Client-side interface 40
5.4 Visualization of land cover class layer 43
5.5 Visualization of confidence assessment layer 44
5.6 Visualization of topographic dataset 45

LIST OF TABLES

2.1 List of OGC web services 10
2.2 DAP requests and responses 14
2.3 List of servers providing input data for hydrodynamic modeling 18

3.1 Metadata basic classes description 22
3.2 Syntax and element Mapping from TDS (THREDDS Dataset Inventory Catalog) to GeoNetwork (ISO19115:2003) 30

5.1 Data discovery mechanism conformance 41
5.2 Data acquisition mechanism conformance 41

Chapter 1
Introduction

1.1 MOTIVATION AND PROBLEM STATEMENT

Inundation (flooding) can be defined as a situation where a large volume of water covers the land surface [50]; this can be caused by various factors such as: heavy rainfall, storm surge, tsunami (earthquake under water) and dam failure. Flooding could result in loss of lives and properties, and if not properly managed it might affect negatively the economy of the country or region in which it occurred.

Real time inundation prediction is a challenge in many areas in the world which lack sufficient data. Three examples of such challenge were the July 2010 Pakistan flood, the May 2011 Colombian flood and the October 2011 Thailand flood–where further predictions could not be made from the current flood extents and timely information could not passed in a consistent manner which resulted into the loss of lives and properties. Effective prediction of real time inundation is a very time and data consuming process, and end users (flood modelers) have to undergo these steps to predict a flood extent:

- Data collection and data preprocessing (data such as Digital Elevation Models and Land cover information).
- Hydrodynamic model parametrization (determining boundary conditions used for the model).
- Hydrodynamic model calibration (involves adjusting the model using certain inputs parameter such as discharge, water level and flood extent images to obtain an optimum level of fit between model predictions and observations).
- Hydrodynamic model validation (evaluating the performance of the model in predicting the observations using the calibrated coefficients).
- Hydrodynamic model prediction.

There exist quite a number of hydrodynamic models, but from the above steps the most time consuming stage for the users (flood modelers) has been the data collection and preprocessing stage since there exist no consistent means to provide these data for areas or regions which lack sufficient data for the models. The ability to quickly predict further flood extents enables emergency personnel's and disaster management teams to make very good decisions in order to avert losses in terms of lives and properties.

Nowadays large amounts of geospatial data are available on the internet as web services. Presently, there are two major web standard protocols for sharing gridded data: the Open-source Project for a Network Data Access Protocol (OPeNDAP) and the Open Geospatial Consortium's (OGC) Web Coverage Service (WCS) standard. OPeNDAP formerly known as the Distributed Oceanographic Data System (DODS) was born in 1995 due to the need for the Earth Science (ES) community to provide the discovery and seamless access to oceanographic data [5]. It uses Data Access Protocol (DAP) which is a data transmission protocol for scientific and environmental data that also allows transmission of gridded data [13]. On the other hand, the Open Geospatial Consortium (OGC) is an international organization founded in 1994 and provides standards for application in different domains to integrate Geographic Information System (GIS) data and services. One of such standard is the Web Coverage Service (WCS) standard which was approved in 2003 [32, 43]. WCS defines standard interface and operations that enables interoperable access to geospatial coverage. The term coverage refers to content data such as satellite images, digital aerial photos, digital elevation model (DEM) data such as SRTM data, and other phenomena represented by values at each measurement point [33].

Both the GIS and ES communities have web standards for the discovery of data needed by users, called catalogue services. The Thematic Realtime Environmental Distributed Data Services (THREDDS) is a data catalogue system developed by Unidata. Unidata is a diverse community made up of over 160 institutions and founded in the early 1990's with a common goal of sharing scientific data, and tools used to access and visualize these data [46]. The main aim of THREDDS is to provide researchers access to a large collection of real-time and archived datasets from various environmental data sources at a number of distributed server sites such as those served by OPeNDAP servers [9]. THREDDS serves as mediator between data providers and data users in order to discover scientific or environmental datasets with the ability to obtain subsets of data which saves time instead of downloading the entire file into their local system [47]. THREDDS is largely used by the ES communities. Parallel to this the OGC's Catalogue Service for Web (CS/W) was designed to support the registry and discovery of geospatial information in order to support the registry and discovery of geospatial information such as those served by WCS. So clients (users) can send a request to a server through HTTP and get a response on metadata information or an error message [34]. OGC's CS/W is largely used by the GIS communities.

Using the above mentioned catalogues, users can be directed to data subsets and gridded data used for inundation prediction but only available as one of the two standards. For inundation prediction, there is the need to use both data from the ES and GIS communities. But there are a number of challenges in interoperable access of metadata as well as unified data discovery between both the Unidata's THREDDS and the OGC CS/W catalogue systems. One of such challenges is the difference in data models (metadata schema) which are served to the users by both catalogue systems [18]. In this research we used web service standards provided by both ES and GIS communities since both are actively involved in different aspects of inundation prediction.

The processes and procedures used to retrieve data and predict flood in data scarce areas are time consuming and needs to be facilitated in order make inundation prediction faster for the flood modelers. From the geo-information perspective, access to input datasets needed for inundation prediction modeling can be facilitated using geo web services. Geo web services can improve access to accurate and complete data needed for inundation prediction modeling. This can be achieved by providing unified (common) access to metadata and by obtaining subsets of data which at the same time will decrease the volume of data that needs processing, reduce computing steps and resources required.

1.2 RESEARCH IDENTIFICATION

The issues that are addressed during this research are identified under the following research objectives and research questions.

1.2.1 Research objective

The main objective of this research is to facilitate an efficient inundation prediction process based on providing unified access to metadata as well as accurate and complete datasets in data scarce areas using geo web services.

1.2.2 Research sub-objectives

From the main research objective stated above, the following sub-objectives were derived:

1. To determine which OPeNDAP and WCS servers serves data needed for inundation prediction modeling.
2. To provide unified access to metadata using both CS/W and THREDDS catalogue servers.
3. To define software stacks for the proposed web service.
4. To determine quality of data accessed by WCS and OPeNDAP servers in terms of accuracy and completeness of the data.
5. To determine the relevant processing routines, this includes resampling of data to predefined grid size. Also define which functionalities/preprocessing on data exist within the above web service standards necessary for inundation prediction modeling.
6. To implement and evaluate a web service that will collect subsets of data as well as preprocessing of data which involves resampling of the collected data for a predefined area.

1.2.3 Research questions

On the basis of research objectives, the following research questions are formulated that should be answered during the research period:

1. How to create an inventory of OPeNDAP and WCS servers which serve data used for inundation prediction modeling?
2. How to consistently provide unified metadata access to the end user through CSW and THREDDS?
3. What software stack (Open source or proprietary or a combination) will be used to implement the web service?
4. How to determine the quality of data accessed by WCS and OPeNDAP servers based on accuracy and completeness of the data?
5. How to preprocess the data? Which functionalities exist within the above web service standards necessary for preprocessing the data, such as which web standard support resampling of data? If such functionalities exist how it is executed within those standards?
6. How to implement the web service(s)?
7. How to evaluate the implemented web service(s)?

1.2.4 Research hypothesis

Geo-information processes that require access to input data needed in hydrodynamic models for inundation prediction modeling (flood extent modeling) as described in section 1.1 can be greatly facilitated using web services. This is based on providing a common data discovery mechanism for these data, speed in which these data are accessed and the provision of complete and accurate datasets for any part of the world.

1.2.5 Innovation aimed at

This research aims at facilitating inundation prediction modeling using web services by providing unified access to metadata using CS/W and THREDDS protocols. In addition, providing access to input data for inundation prediction modeling using WCS and OPeNDAP protocols.

1.2.6 Related work

Efforts have been made to ascertain interoperability between OPeNDAP data access protocols and the OGC WCS. One of the approaches was based on THREDDS Data Server (TDS), designed by Unidata to combine implementations of THREDDS catalogue services with integrated data serving capabilities, including OPeNDAP and WCS [6]. But in this interoperability approach, there were issues on metadata incompatibilities that were addressed by transforming metadata used by OPeNDAP standard to that of WCS. The approach is one-directional since it emphasizes OPeNDAP clients having access to WCS servers, but not the other way around.

Similarly, the National Aeronautics and Space Administration (NASA) in response to the need of the World Climate Research Program (WCCRP) [23] tried to develop a bi-directional gateway that would allow an OPeNDAP client to access the data and services provided by an OGC compliant WCS server and also for a WCS client to access an OPeNDAP server. It is called the two-way interoperability between the OPeNDAP and WCS clients and servers. The interoperability gateway is not generic and it is yet to be fully implemented. The project did not fully address interoperability in the discovery of datasets, which means it failed to build a catalogue gateway that would give access to metadata used in both GIS and Earth Science communities.

An experiment was done to clarify the semantic and syntax interoperability between the OGC CS/W and the THREDDS catalogue systems in order to try to bridge the gap between both protocols [18]. This work focused on fostering interoperability between OGC CS/W and THREDDS with an effort to information model transformation by CS/W ISO profile mapping with THREDDS catalogue schema. The research discussed challenges in general interoperability between both catalogue systems.

The above shows various efforts in bridging the gaps in data discovery between THREDDS and CS/W, and data access between OPeNDAP and WCS. Despite these efforts, there are still a number of issues especially in the field of inundation prediction. This research will be looking into issues in serving datasets from WCS and OPeNDAP protocols through CS/W and THREDDS catalogue servers; I will research ways of giving access to metadata information and how both catalogue systems can be used in order to have a unified access to predefined datasets that will be used as inputs in hydrodynamic models for inundation prediction.

1.3 PROJECT SET-UP

This section describes the approach that was used to answer the research questions as shown in figure 1.1

1.3.1 Method adopted

The research began with finding out user requirements for inundation prediction modeling (flood extent modeling), the kind of data required, a comprehensive review of web servers that provided such data and the quality of data accessed.

The various OPeNDAP and WCS servers was reviewed. In order to ascertain the kinds of data is obtainable and to find out which data preprocessing capabilities existed on the server side appropriate for inundation prediction modeling. WCS servers were chosen because they are most appropriate OGC web service for accessing grid coverage's in the same way OPeNDAP servers were chosen because they have the ability to obtain subsets of files, and also the ability to aggregate data from different files.

A data discovery mechanism was designed and implemented in order to gain access to flood related data through CS/W and THREDDS catalogue servers, and provide a rich metadata of datasets for flood related data. This mechanism was evaluated based on functionality (spatial/temporal/attribute subsetting).

In addition, a prototype web service for data acquisition was designed and implemented. This involved retrieving datasets using freeware Python libraries to access OPeNDAP and WCS servers. The prototype web service for data acquisition was evaluated based, firstly on functionality, secondly on performance (based on speed in which the data is accessed) and thirdly on accuracy and completeness of data accessed.

In the final stage of the research I conceptually designed a loosely coupled web service infrastructure which was composed from the previous two independent mechanisms. This infrastructure served as proof of concept demonstration for an implementable web service infrastructure that could be used to discover and retrieve datasets that would serve as inputs for inundation prediction modeling. The web service infrastructure was evaluated independently based on the evaluation of the previous two mechanisms. The applicability of the infrastructure for users was analyzed based on the accuracy and completeness of the data provided.

1.4 ORGANIZATION OF THESIS

The remaining chapters of this thesis are organized as follows:

chapter 2: This chapter gives a description of users, users profile and the users requirements on flood related data. It describes characteristics of flood related data with an inventory of existing web services providing some of these data. In addition, the chapter gives a brief introduction to web services for both GIS and ES communities and literature on issues of interoperability between the two communities as well as existing overview of techniques used for visualizing flood extent maps based on satellite images.

chapter 3: This chapter deals with issues of interoperability between the GIS and ES communities, and describes a conceptual framework for giving unified access to data. As well as a description of challenges for interoperability.

chapter 4: This chapter describes the design and implementation of a prototype web service for retrieving flood related data. The chapter discusses the choice of clients to access these data and the challenges faced during its implementation.

chapter 5: This chapter gives a brief conceptual design for an infrastructure that encompasses mechanism from the previous two chapters. This chapter forms the core for a description of a web service infrastructure that will be used to discover and retrieve dataset used for flood extent modeling. The chapter also described evaluating the web infrastructure based on implementation from the previous two mechanisms.

chapter 6: This final chapter gives answers to the research questions, discusses on the findings of the research, and suggests recommendations for further research.

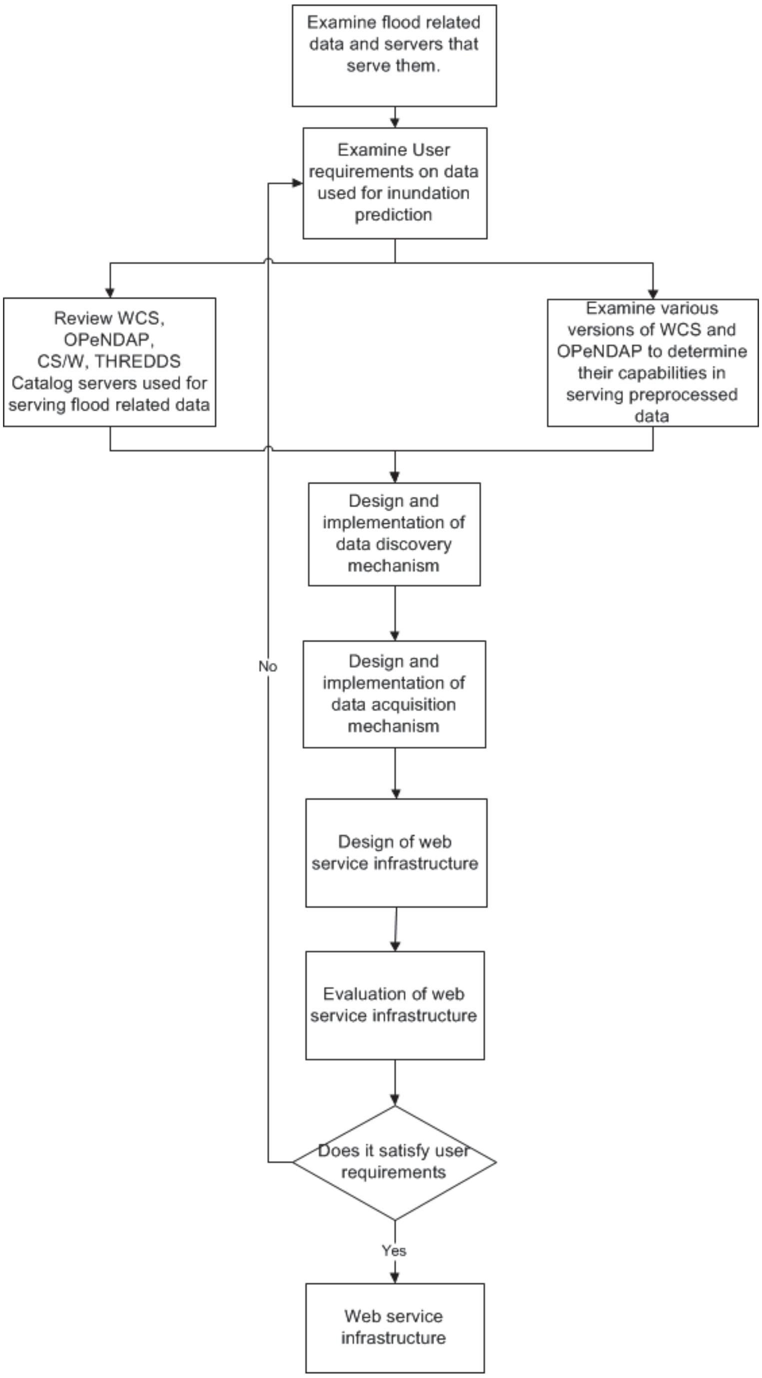


Figure 1.1: Research framework to show method adopted

Chapter 2
Literature review and user requirements

2.1 USERS AND USERS REQUIREMENTS FOR REAL-TIME INUNDATION PREDICTION MODELING

From the perspective of this research, users that perform real time inundation prediction are the flood modelers. Users (flood modelers) require faster means of accessing their data needed for their inundation prediction model in order to produce a flood extent map which is later used for decision making. Furthermore the quality of data accessed as well as speed in which it is accessed plays an important role in the accuracy of flood extent map (inundation map) produced. The following assumptions will be made on my user profile:

1. Users have background knowledge of the input datasets used in their hydrodynamic model.
2. Users do not have the required skills to determine which web services provide faster access to these input datasets as well as processing capabilities provided by these web services necessary for hydrodynamic model.
3. Users have data finding issues, issues related to how to consistently find necessary data from a large variety of data sources considering the fact that study area is not always the same.
4. Users require mechanism that allows them to search for data using spatial, temporal or attribute subsets. They also require a mechanism that allows them to preprocess the data before it serves as input for their hydrodynamic models.

2.1.1 Users requirements on data used for inundation prediction modeling and their characteristics

The data collection and preprocessing phase of inundation prediction modeling is the first and the most time consuming stage for the users to predict flood extents. This is due to the fact that there exist no consistent means to discover and access fit for use datasets needed for flood extent modeling. Provision of a consistent and timely system that can provide data that serves as input parameter for hydrodynamic models will hasten inundation prediction which in turn will lead to timely and proper decision making.

Users require dataset that will serve as input parameters for a hydrodynamic model in order to get a flood extent map, for a two-dimensional(2D) hydrodynamic model. These input data needed for parameterization are described below:

1. An accurate Digital Elevation Model (DEM) of the area to be simulated is required. But usually in data scarce areas, the Shuttle Radar Topography Mission (SRTM) data is used as a DEM source since it has a global coverage. SRTM DEM gives a near-global elevation and covers the Earth between latitudes 60N and 57S has a spatial resolution of 30 meters for the United States of America (USA) and a spatial resolution of 90m outside the USA. SRTM has an absolute vertical accuracy of 16 meters and an absolute horizontal accuracy of 20 meters [42, 21]. Another source of global DEM is the Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) Global Digital Elevation Model Version 2

(GDEM V2). On October 17, 2011 it was jointly released by The Ministry of Economy, Trade, and Industry (METI) of Japan and the United States National Aeronautics and Space Administration (NASA), and with a global spatial resolution of 30 meters [3]. DEM's are mostly used in hydrodynamic models to extract the cross-section elevations of rivers and river networks as well as bathymetry of rivers especially in regions where data is scarce [40].

2. Land cover information of the area in question. Land cover classes are used to determine friction factor coefficients as a parameter in the hydrodynamic model [17]. Land cover classes can be derived from either: (1) the Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) images of between 15 - 30 meters spatial resolution [4, 51]. (2) Moderate Resolution Imaging Spectroradiometer (MODIS) land products with spatial resolutions of 500 meters [7].(3) LandsatTM (Thematic Mappers) images can also serve as a source of providing land cover classes, it has a spatial resolution of 30 meters in 6 out of its 7 bands.

3. Boundary conditions (river discharge and water depth at a particular time series). Boundary conditions which includes: Discharge (Q) and water level (h) at both upstream and downstream boundary of the model respectively are usually obtained from gauging stations.

4. Flood extent data (flood extent data can be obtained from processed Envisat Advanced Synthetic Aperture Radar (ASAR) images characterized with a pixel size of 75m and with approximately 150 meters ground resolution) are used for calibration and validation of the model [8]. The LandsatTM is another source of data for deriving flood extent as shown by various studies [19, 20] and can as well serve as calibration data.

5. Line elements which are characterized as vector data (such as raised roads, culverts and embankments) which are not visible from the DEM data at a particular spatial resolution. These are used based on adequate spatial information of such elements above the terrain in the location of interest or if there exist assumptions on assigning raised value for each type of these element. They are mostly used in the hydrodynamic model at the same scale as that of the model. One source of line elements is OpenStreetMaps (OSM) [12] which is an initiative for the creation, visualization and provision of geographic data.

These data which are characterized above are usually all brought to the same grid size depending on the hydrodynamic model, and then serve as input parameters for various 2D numerical hydrodynamic models as shown in figure 2.1 (the grey areas in the figure 2.1 shows areas that will not be discussed because it is beyond the scope of this research) that will produce future flood extent maps based on current flood extent. The accuracy of the flood extent maps are based on the accuracy of these data as well as completeness of the data.

Hydrodynamic models are numerical models that solve governing equations for flow in rivers and floodplains. These models are categorized by [40] as follows: (a) One Dimensional (1D) models. (b) two-dimensional(2D) models and (c) 1D river flow models coupled with 2D floodplain flow (1D–2D) models. This study focuses more on 2D raster routing models (e.g LISFLOOD–FP) which is a type of 2D model, this is because these type of models use the raster DTM to schematize the floodplain as a regular grid with each pixel in the grid treated as an individual storage cell [44]. Such models are known to be sensitive to scale dependency that is different grid sizes yields different output results, smaller grid sizes gives more accurate results but requires more computational time to produce output, while larger grid sizes does not highlight salient features/profiles but is computationally faster.

Studies [16, 21] showed that these models are also sensitive to the resampling and resampling methods on input data. Resampling of high resolution data always leads to a great loss of infor-

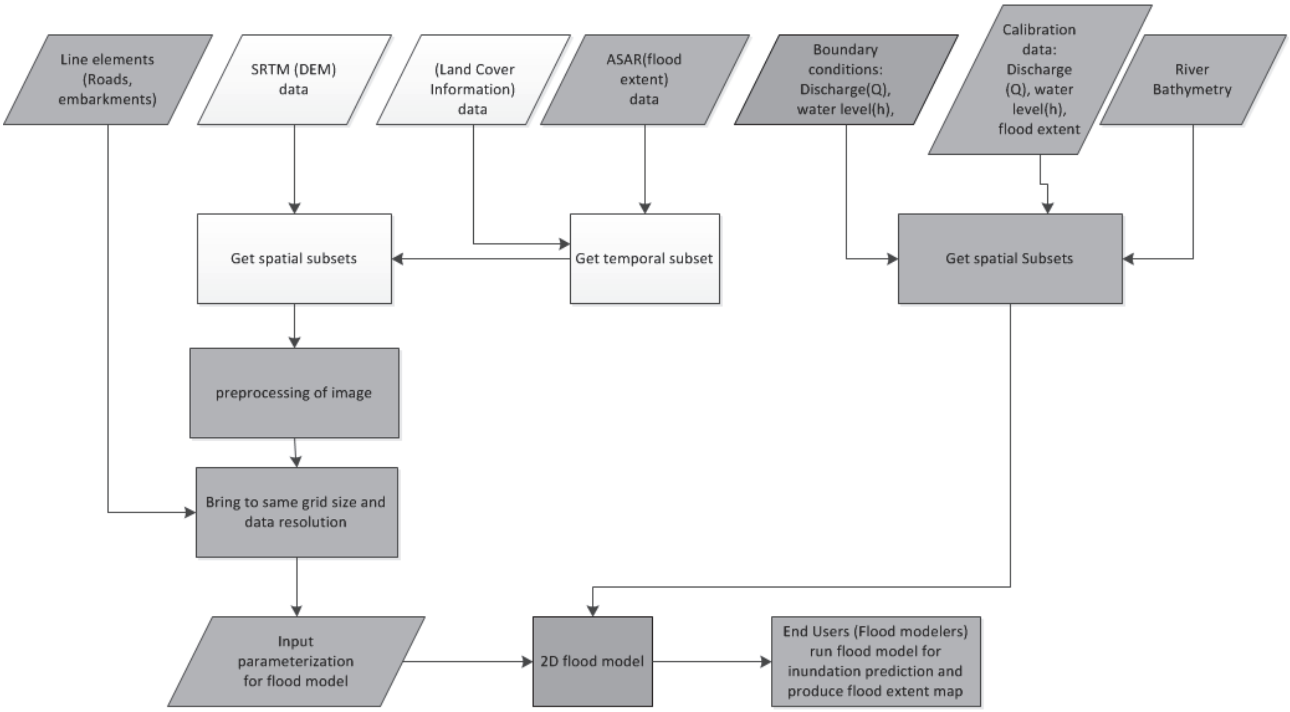


Figure 2.1: User requirements for hydrodynamic modeling

mation and thereby accounts for errors in output of the models. Therefore depending on the requirements of the hydrodynamic model great care should be taken into choosing the level of resolution as well as resampling methods on the input data for the model.

Many data providers make use of the internet to provide some of these input data mentioned and characterized above. These data are made available as web services. But before we delve into details of describing the sources of these data let us briefly define web services.

2.2 WEB SERVICES

Nowadays the internet houses large amounts of data and these data are available to us as web services. A web service is defined by [15] as "an interface that describes a collection of operations that are network-accessible through standardized XML messaging". We can have access to geospatial data as well as processing functionalities using open standards through web services.

2.3 OGC WEB SERVICES

The Open Geospatial Consortium (OGC) is an international consortium of 432 companies, government agencies and universities. OGC was founded in 1994 and provides standards for application in different domains to integrate Geographic Information System (GIS) data and services [32]. OGC provides services with standardized interfaces. Some of these services are listed in the table 2.1.

The Service-Oriented Architecture (SOA) as shown in figure 2.2 serves as the base architecture for which these standards are implemented. Applications built based on some of these OGC standards can provide some of the data needed for hydrodynamic models in order to predict flood extents. Two of these OGC standards that provide access to data are the WCS and the WFS which

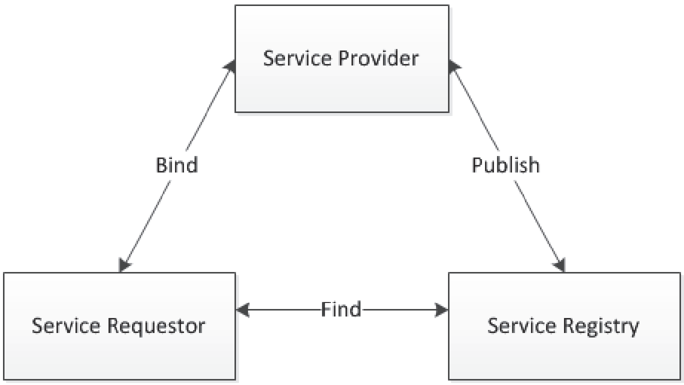


Figure 2.2: Service-Oriented Architecture (SOA) diagram

Table 2.1: List of OGC web services

Service Name	Service Description
Web Map service (WMS)	Displays an image (gif, jpeg, png) of vector or raster data
Web feature Service (WFS)	Provides access to vector data
Web Coverage service (WCS)	Provides access to raster data
Web Processing service (WPS)	Provides access to remote GIS Operations for example Buffer, Distance calculation, Coordinate transformation
Catalog Service for the Web (CS/W)	Provides access to metadata of geospatial data and services

are highlighted below and in relation to this the OGC CS/W standard that provide metadata for geospatial data and services will be highlighted as well.

2.3.1 Web Coverage Service (WCS)

The OGC WCS supports the electronic retrieval of geospatial data as "coverages" that is, digital geospatial information representing space/time-varying phenomena. A WCS provides access to coverage data in forms that are useful as input into scientific models and for other clients. The term coverage refers to content data such as satellite images, digital aerial photos, digital elevation data such as SRTM data, and other phenomena represented by values at each measurement point [33].

WCS allows clients to query for subsets of data on its server. Such subsets include spatial and temporal subsets. In addition, WCS also allows for reprojection of geospatial data. These properties make WCS similar with OGC Web Map service (WMS) and Web Feature Service (WFS). But "Unlike WMS, which portrays spatial data to return static maps (rendered as pictures by the server), the Web Coverage Service provides available data together with their detailed descriptions; defines a rich syntax for requests against these data; and returns data with its original semantics (instead of pictures) which may be interpreted, extrapolated, etc., and not just portrayed. Unlike WFS, which returns discrete geospatial features, the Web Coverage Service returns coverages representing space/time-varying phenomena that relate a spatio-temporal domain to a (possibly multidimensional) range of properties. As such, WCS focuses on coverages as a specialized class of features and, correspondingly, defines streamlined functionality" [29].

The WCS performs three basic operations, they are: GetCapabilities, DecribeCoverage and GetCoverage. These operations are followed by their responses which are GetCapabilities response, DescribeCoverage response and GetCoverage response respectively [25].

1. GetCapabilities: The GetCapabilities operation allows WCS clients to access service meta-

data from a WCS server. For which the response is an XML document describing the server capabilities.

2. DescribeCoverage: The DescribeCoverage operation gives a description of one or more available coverages in a WCS server in XML format as requested by the WCS client. In response to a DescribeCoverage request, WCS returns XML document whose top-level element is a CoverageDescription containing CoverageOffering elements for all coverages requested. CoverageOffering extends CoverageOfferingBrief, to provide additional details on the domain and range of a coverage offering. Clients may use these to assess the data's fitness for use, and to formulate fine-grained GetCoverage requests [25].

3. GetCoverage: The GetCoverage operation allows for the retrieval of coverages. It is a mandatory operation for all WCS clients and servers. GetCoverage operation is performed after GetCapabilties and DescribeCoverage operations. This operation allow spatial, temporal, band subsetting, scaling, reprojection, and final result packaging, including data format encoding. One GetCoverage operation returns single coverage at a time that is encoded in a well-known coverage format like HDF-EOS, NITF, and GeoTIFF. The normal response to a valid GetCoverage operation request shall be a single coverage extracted from the coverage requested, with the specified spatial reference system, bounding box, size, format, and range subset [28].

A WCS should be able to provide the following functionalities when implemented:

1. Spatial query (by specifying its BBOX)
2. Temporal query (by specifying time)
3. Resampling (by specifying Interpolation method and resx, resy parameters)
4. Reprojection (by specifying CRS)
5. Output format (by specifying its output)

The above functionalities provided by the WCS server are necessary for processing the input data before they are used in hydrodynamic models for inundation prediction modeling.

2.3.2 Web Feature Service (WFS)

The OGC WFS allows a client to retrieve and update discrete geospatial data encoded in Geography Markup Language (GML) from multiple Web Feature Services. "The WFS operations support INSERT, UPDATE, DELETE, LOCK, QUERY and DISCOVERY operations on geographic features using HTTP as the distributed computing platform" [26]. The WFS standard defines operations that enable clients to:

1. GetCapabilities: A web feature service must be able to describe its capabilities.
2. DescribeFeatureType: A web feature service must be able, upon request, to describe the structure of any feature type it can service.
3. GetFeature: A web feature service must be able to service a request to retrieve feature instances.
4. GetGmlObject: A web feature service may be able to service a request to retrieve element instances by traversing XLinks that refer to their XML IDs.

5. Transaction: A web feature service may be able to service transaction requests. A transaction request is composed of operations that modify features; that is create, update, and delete operations on geographic features.

6. LockFeature: A web feature service may be able to process a lock request on one or more instances of a feature type for the duration of a transaction.

Based on the Operations defined above, three kinds of WFS can be defined [26]:

1. Basic WFS: A basic WFS would implement the GetCapabilities, DescribeFeatureType and GetFeature operations. This would be considered a READ-ONLY web feature service.

2. Xlink WFS: An XLink WFS would support all the operations of a basic web feature service and in addition it would implement the GetGmlObject operation for local and/or remote XLinks, and offer the option for the GetGmlObject operation to be performed during GetFeature operations.

3. Transaction WFS: A transaction web feature service would support all the operations of a basic web feature service and in addition it would implement the Transaction operation. Optionally, a transaction WFS could implement the GetGmlObject and/or LockFeature operations.

From the different kinds of WFS, the Basic WFS can serve as a data source for the input of possible line elements such as raised roads, culvert and embankments for the hydrodynamic model since it implements the basic WFS operations needed to retrieve discrete data.

2.3.3 Catalogue Service for the Web (CS/W)

The OGC CS/W is a web service which provides access to metadata of geospatial data and services. It serves as the link between the service requestor and the service provider. The CS/W supports discovery and binding to geo web services. CS/W has five mandatory request operations which are: GetCapabilities, DescribeRecord, GetDomain, GetRecords and GetRecordsById. These operations are further classified into service operations comprising the *GetCapabilities operation* and the *discovery operations* (DescribeRecord, GetDomain, GetRecords and GetRecordsById). CS/W equally has two optional request operations which are *transaction request* and *harvest request* and both are classified as management operations [34]. The sequence of service operation and response is shown in figure 2.3

The CS/W will serve as a communicator of data quality for the users who needs this information for the data that will serve as input for hydrodynamic models.

2.4 OPENDAP

The Open-source Project for a Network Data Access Protocol (OPeNDAP) formerly known as the Distributed Oceanographic Data System (DODS) was born in 1995 due to the need for the Earth Science community to provide the discovery and seamless access to oceanographic data [5]. It uses Data Access Protocol (DAP) which is a data transmission protocol for scientific and environmental data that also allows transmission of gridded data.

DAP uses three responses to represent a data source. They are the Dataset Descriptor Structure (DDS) and Dataset Attribute Structure (DAS), the Data Dataset Descriptor Structure (DataDDS). The first two responses characterize the variables, their datatypes, names and attributes. While the third response holds the data values along with name and datatype information. DAP equally

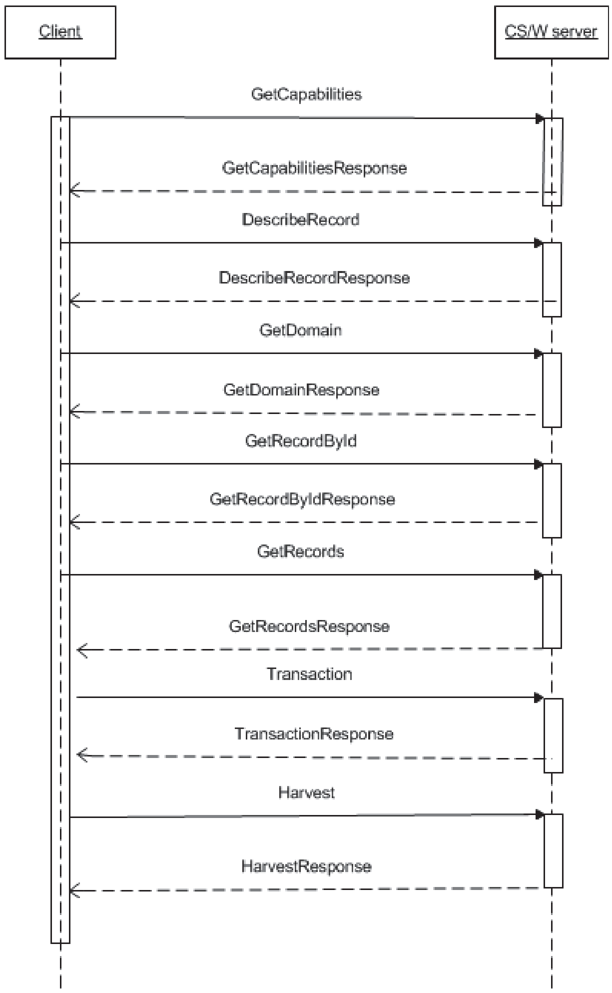


Figure 2.3: Basic CS/W operations

Table 2.2: DAP requests and responses

Request	Response
DDS	Dataset Descriptor Structure or Error
DAS	Dataset Attribute Structure or Error
DataDDS	Data Dataset Descriptor Structure or Error
Server version	Version information as text
Help	Help text describing all request-response pairs

returns an error response if a request for any of the three basic responses cannot be returned. DAP request and response is summarized in table 2.2. Thus, the use of DAP effectively isolates the format and structure from the data itself and at the same time allows for spatial, temporal and parameter subsetting depending on the data model employed as well as projection operations on the data. Example: a user requests a subset of data from a remote archive via a URL. The server extracts the subset of interest, transforms this subset to the OPeNDAP data model, compresses the resulting data object, and sends it to the requesting client. On receipt, the client decompresses the data stream, translates it from the OPeNDAP data model to the data model used by the client application, and enters it into the application's workspace. [13, 5]. Spatial and temporal subset are mostly based on data array subset, which means subset range is specified by using array index, Subsets of data can equally be gotten using spatial extents (latitude and longitude) and temporal extent (time or time difference). OPeNDAP has various clients in which its users can have access to OPeNDAP data as well as display its data. OPeNDAP clients some of which includes [37]: Matlab Command-line client, Ferret, OPeNDAP Matlab Toolkit, GrADS, Integrated Data Viewer(IDV), Generic Web Browser.

The OPeNDAP server makes available various file formats such as Grib, HDF (Hierarchical Data Format) and netCDF (network Common Data Form) available through the DAP protocol.

2.5 THREDDS

The Thematic Realtime Environmental Distributed Data Services (THREDDS) is a data catalogue system developed by Unidata. Unidata is a diverse community made up of over 160 institutions and founded in the early 1990's with a common goal of sharing scientific data, and tools used to access and visualize these data [46]. The main aim of THREDDS is to provide researchers access to a large collection of real-time and archived datasets from various environmental data sources at a number of distributed server sites such as those served by OPeNDAP servers [9].

THREDDS serves as mediator between data providers and data users in order to discover scientific or environmental datasets with the ability to obtain subsets of data which saves time instead of downloading the entire file into their local system. THREDDS Dataset Inventory Catalogs are used to provide virtual directories of available data and their associated metadata. These catalogs can be generated dynamically or statically [47].

THREDDS has various clients in which its users can have access to data and metadata as well as display its data. THREDDS clients include [9]:

1. THREDDS Thin Clients:

    - Live Access Server (LAS, PMEL, Steve Hankin). LAS illustrates the use of a Web-based (thin) client with the bulk of the analysis and display generation done on the server side.

- Ingrid (IRI/LDEO, Benno Blumenthal). This is another example of a system enabling analysis and display of data via a Web browser.
- GDS, GrADS/DODS Server (COLA, Center for Oceans Land Atmosphere, Joe Wielgosz)

2. THREDDS Thick Clients:

   - IDV (Integrated Data Viewer). Meteorological Applications MetApps (Unidata Program Center, Don Murray). A set of pure Java, platform-independent, two- and three-dimensional data-analysis and display tools-based on the VisAD infrastructure.
   - Others: Some software packages such as MatLab, Interactive Data Language (IDL), Man-computer Interactive Data Access System (McIDAS) have already been adapted to acquire remote data via DODS or ADDE. Even if these systems are not adapted to take direct advantage of Catalogs or other THREDDS advances.

The combination of both the OGC CS/W and THREDDS catalogue servers will serve as the basis for providing metadata on datasets used as input parameters for the hydrodynamic models.

2.6 PREVIOUS WORK ON DATA PROVISION THROUGH WEB SERVICES

There has been efforts to ascertain interoperability between OPeNDAP data access protocols and the OGC WCS. One of the approaches was based on THREDDS Data Server (TDS), designed by Unidata to combine implementations of THREDSS catalogue services with integrated data serving capabilities, including OPeNDAP and WCS [6]. But in this interoperability approach, there were issues on metadata incompatibilities that were addressed by transforming metadata used by OPeNDAP standard to that of WCS. The approach is one-directional since it emphasizes OPeNDAP clients having access to WCS servers, but not the other way around.

Furthermore, the National Aeronautics and Space Administration (NASA) in response to the need of the World Climate Research Program (WCCRP) [23] tried to develop a bi-directional gateway that would allow an OPeNDAP client to access the data and services provided by an OGC compliant WCS server and also for a WCS client to access an OPeNDAP server. It is called the two-way interoperability between the OPeNDAP and WCS clients and servers. The interoperability gateway is not generic and it is yet to be fully implemented. The project did not fully address interoperability in the discovery of datasets, which means it failed to build a catalogue gateway that would give access to metadata used in both GIS and Earth Science communities.

More work was done to clarify the semantic and syntax interoperability between the OGC CS/W and the THREDDS catalogue systems in order to try to bridge the gap between both protocols [18]. This work focused on fostering interoperability between OGC CS/W and THREDDS with an effort to information model transformation by CS/W ISO profile mapping with THREDDS catalogue schema. Similar work was done to ingest dataset records from THREDDS data server into GeoNetwork CS/W server [49]. Both researches discussed challenges in general interoperability between both catalogue systems. My research will be based on using both methods in order to aid interoperability between OGC CS/W and THREDDS as it relates to finding datasets that are required for inundation prediction modeling only.

There has also been an effort to combine OPeNDAP and WCS protocols by providing a web service that matches trajectory data and raster data. The middleware of the web service consisted of OWSLib Python library which can access remote data sources using OGC specification and also Pydap Python library implemented on OPeNDAP as clients to access scientific dataset efficiently as well as a server to distribute data in different formats [22].

The Atmospheric Data Discovery System (ADDS) infrastructure was developed to provide an efficient data discovery environment for observational datasets in the atmospheric sciences and with the ability to obtain subsets of data [39]. The ADDS infrastructure will also enable users to obtain subsets of data from large datasets. The framework of ADDS comprises: (1) a portal server, (2) a metadata cataloguing engine server, (3) A relational database server, (4) a data storage for temporary data. But this research was constrained to only the discovery of observational data in the atmospheric sciences and also constrained to a single data format used known as Binary Universal Form (BUFR).

Recently OGC announced that the netCDF is added to the OGC standard for communicating multidimensional data. netCDF was originally developed for the Earth science community, and netCDF can be used to store a variety of multidimensional data. It is a self-documenting, which means it can internally store information used to describe the data. This encoding format is assumed to aid data interoperability as well as data integration between both Earth Science and GIS communities [30].

OGC recently completed a Water Information Services Concept Development Study Engineering Report, which provides guidelines for open information system architectures that support publishing, cataloguing, discovering and accessing water observations data using open standards. This will help users such as research organization, universities, and government bodies to discover, access and integrate data from multiple sources in studies related to hydrological science and water resources management [31]. In relation to this, OGC completed a Oceans/Meteorological/Hydrology Water Cycle summit where the Open Geospatial Consortium Hydrology Domain Working Group (DWG) and the OGC Meteorology and Oceanography Domain Working Group worked hand in hand to solve issues of interoperability and easy access between hydrological, oceanographic, meteorological and climatological related information. This summit brought about the creation of the WaterML 2.0 Standards Working Group (SWG), and is about to be a candidate to be adopted internationally as an OGC standard [35].

Currently, there exist production of flood extent maps from satellite images such as MODIS and Envisat ASAR, one of such initiatives is the Unitar's Operational Satellite Applications Programme (UNOSAT). The images are usually analysed using ESRI ArcGIS but lack field validation [48]. Similarly, another institution known as the Dartmouth flood observatory founded in 1993 at Dartmouth College, Hanover, NH USA and moved to the University of Colorado in 2010. They are responsible for facilitating practical use of space-based information for international flood detection, flood response, future risk assessment, and hydrological research. This space based information includes the using of both MODIS sensors images obtained aboard the Terra and Aqua spacecrafts [24]. But flood extent maps produced by these institutions cannot serve as basis for the prediction of next inundated areas. Its limitation is that it can only show in near-real time current inundated areas.

The above literature shows various efforts in bridging the gaps in data discovery between THREDDS and CS/W, and data access between OPeNDAP and WCS. In addition, the review reveals existing techniques used for visualizing flood extent maps based on satellite images. However, there are still a number of challenges especially in the field of inundation prediction. One such challenge is an integrated means of providing all the necessary data used as input parameters for inundation prediction modeling as well as providing their metadata. Furthermore, this study will be looking into issues in serving datasets in WCS and OPeNDAP services through CS/W and THREDDS catalogue servers; and exploring ways of giving access to metadata information and how both catalogue systems can be used in order to have a unified access to predefined datasets that will be used as input parameters in hydrodynamic models for inundation prediction modeling.

2.7 OVERVIEW OF WCS, OPENDAP, CS/W AND THREDDS SERVERS HAVING FLOOD RELATED DATA

A comprehensive review was done to ascertain which open (free) servers that contained input data needed for inundation prediction modeling irrespective of spatial coverage, for the THREDDS/OPeNDAP servers we reviewed by manually checking a THREDDS client IDV as well as checking OPeNDAP current dataset list [38] for available servers providing them. Similarly, I checked for free CS/W and WCS servers serving datasets used for inundation prediction modeling by going through OGC's compliant product and services. An inventory of servers serving these datasets are listed in table 2.3

2.8 CONCLUSION

This chapter described the users, user's profile and assumptions made based on the user. It described users requirements and expectations on data served as input parameters for flood extent modeling. In addition, the different web servers were described and it was found that the OGC WCS server can implement resampling and reprojection of data which are required data processing functionalities for inputs in hydrodynamic models. The chapter also gave an overview on previous work on data provision through web services. In addition we provided a inventory of different web services that serves these input data from both GIS and ES communities. It should however be noted that from the inventory in table 2.3, there are currently no available web service that provides flood extent data (for example ASAR data). Likewise, there is no existent web service that provides calibration data (discharge and water level). Without these missing data, users cannot carry out inundation prediction modeling.

Table 2.3: List of servers providing input data for hydrodynamic modeling

Server	data	URL	Coverage
WCS (Geobrain _WCS_DEM)	SRTM global (90m), GTOPO30arc Global	http://geobrain.laits.gmu.edu/ cgi-bin/gbwcs-dem?service=wcs &version=1.0.0 &request=getcapabilities	Global
ORNL DAAC WCS Server	Global AVHRR , land cover, MODIS land cover	http://webmap.ornl.gov/ ogcbroker/wcs?service= WCS&version=1.0.0 &request=GetCapabilities	Global
GMU-LAITS CS/W	SRTM 90m Global, LandSAT, ASTER DEM,	http://geobrain.laits.gmu.edu /GeoDataDownload	Global
FAO GeoNetwork CS/W	MODIS data, land cover, SRTM 90m Global	http://www.fao.org/geonetwork /srv/en/main.home	Global
ERDDAP Server (OPeNDAP)	SRTM30+ version 6.0 (30 arc second)	http://coastwatch.pfeg.noaa.gov/ erddap/griddap /usgsCeSrtm30v6.html	Global
GrADS Data Server (OPeNDAP)	Land cover	http://www.monsoondata.org: 80/dods/landcover	Global
Oceanwatch-Cental Pacific THREDDS Server	Smith and Sandwell v8.2 -Topography and Bathymetry	http://oceanwatch.pifsc.noaa.gov /thredds/dodsC/bathymetry/smith _sandwell_topo_v8_2.nc	Global
ORNL DAAC THREDDS Data Server	ISLSCP-2 Global Land Cover Class (hd) 0.25 degree resolution	http://thredds.daac.ornl.gov /thredds/dodsC/968/Land _Cover_Class_0.25degree.nc	Global
ORNL DAAC THREDDS Data Server	Land Cover Class 0.5degree (MODIS Land Cover) 0.5 degree resolution	http://thredds.daac.ornl.gov /thredds/dodsC/968/Land _Cover_Class_0.5degree.nc	Global
USGS Woods Hole THREDDS Data Server 1	SRTM30+ Version 1.0 (30 arc second - Worldwide)	http://geoport.whoi.edu/ thredds/dodsC/ bathy/srtm30plus_v1.nc	Global
USGS Woods Hole THREDDS Data Server 1	Smith and Sandwell v9.1 (60 arc second - Worldwide)	http://geoport.whoi.edu /thredds/dodsC/bathy /smith_sandwell_v9.1.nc	Global
Rutgers THREDDS Data Server	srtm bathymetry smith_sandwell	http://tashtego.marine. rutgers.edu:8080 /thredds/catalog/other/bathymetry catalog.html	EasternUSA

Chapter 3

Design and implementation for common data discovery mechanism

The previous chapter demonstrated various issues and efforts to bridge the gap of interoperability issues between GIS and ES communities. Despite these efforts, the gap still exists. This chapter looks specifically at interoperability issues related to data discovery and metadata search for input data for flood extent modeling. It also discusses the challenges and provides a proof of concept demonstration and implementation for giving consistent access to metadata needed for flood extent modeling.

3.1 INTEROPERABILITY ISSUES BETWEEN THREDDS AND OGC CS/W

There is need for interoperability between users of both GIS and ES communities to make data accessible from both communities but the difference in metadata schema between THREDDS and CS/W is a major constraint. In the sections below we describe the metadata schema of THREDDS and CS/W and highlight the differences between both schemas.

3.1.1 Metadata schema for THREDDS catalogue

The THREDDS catalogue is made up of Extensible Markup Language (XML) files and is compliant with the THREDDS Dataset Inventory Catalog Specification 1.0.2 which is used to give structures for the organization of datasets, access methods for each datasets and a name. THREDDS catalogs are XML documents that can describe both directory and inventory level information for a data source. A summary of the basic catalog elements are listed below as specified in [45]:

- Base Catalog Elements:
    1. Catalog: The *catalog* element is the top-level element. It may contain zero or more service elements followed by zero or more property elements followed by one or more *dataset* elements (actually any element in the dataset substitution group: dataset or catalogRef).
    2. Service: A *service* element represents a data access service and allows basic data access information to be factored out of *dataset* and *access* elements.
    3. Dataset: A *dataset* element represents a named, logical set of data at a level of granularity appropriate for presentation to a user. A dataset is *direct* if it contains at least one dataset access method, otherwise it is just a container for nested datasets, called a *collection* dataset. The name of the dataset element should be a human readable name that will be displayed to users. Multiple access methods specify different services for accessing the same dataset.
    4. Access: An *access* element specifies how a dataset can be accessed through a data *service*. It always refers to the dataset that it is immediately contained within.

5. catalogRef: A *catalogRef* element refers to another THREDDS catalog that logically is a nested dataset inside this parent catalog. This is used to separately maintain catalogs and to break up large catalogs. THREDDS clients should not read the referenced catalog until the user explicitly requests it, so that very large dataset collections can be represented with catalogRef elements without large delays in presenting them to the user.
6. XLink: The *XLink specification* that are used to point to another web resource. The *xlink:href* attribute is used for the URL of the resource itself. The *xlink:title* attribute is a human-readable description of the linked resource. THREDDS clients can display the title to the user as appropriate. These are the only two attributes currently used in the THREDDS software. You can also add the *xlink:type* or *xlink:show* attributes.

- Digital Library Metadata Elements: These are catalog elements that are used in Digital Libraries entries, discovery centers, and for annotation and documentation of datasets.

1. threddsMetadataGroup: The elements in the *threddsMetadataGroup* may be used as nested elements of both dataset and metadata elements. There may be any number of them in any order, but more than one geospatialCoverage, timeCoverage, dataType, dataFormat, serviceName, or authority elements will be ignored.
2. documentation: The *documentation* element may contain arbitrary plain text content, or XHTML.We call this kind of content "human readable" information. It has an optional documentation type attribute, such as summary, funding, history, etc. The documentation element may also contain an XLink to an HTML or plain text web page.
3. metadata: A *metadata* element contains or refers to structured information (in XML) about datasets, which is used by client programs to display, describe, or search for the dataset. We call this kind of content "machine readable" information.
4. property: Property elements are arbitrary name/value pairs to associate with a *catalog*, *dataset* or *service* element. Properties on datasets are added as global attributes to the THREDDS data model objects. More specialized semantics will be defined in the future.
5. sourceType: This is used by the creator and publisher elements to specify who is responsible for the dataset. It must have a name and contact element.
6. contributor: A *contributor* is simply a person's name with an optional role attribute that specifies the role that the person plays with regard to this dataset.
7. geospatialCoverage: A *geospatialCoverage* element specifies a lat/lon bounding box, and an altitude range that the data covers.
8. timeCoverage: A *timeCoverage* element specifies a date range.
9. dateType: A *dateType* follows the W3C profile of ISO 8601 for date/time formats. Note that it is a simple type, so that it can be used as the type of an attribute.
10. dateTypeFormatted: A *dateTypeFormatted* extends dateType by allowing an optional, user-defined format attribute and an optional type attribute.
11. A duration type can be one of the following:
    (a) an xsd:duration type specified in the following form "PnYnMnDTnHnMnS" where:
        - P indicates the period (required)
        - nY indicates the number of years

- nM indicates the number of months
- nD indicates the number of days
- T indicates the start of a time section (required if you are going to specify hours, minutes, or seconds)
- nH indicates the number of hours
- nM indicates the number of minutes
- nS indicates the number of seconds

(b) a valid udunits time duration string.

12. dataSize: A *dataSize* element is just a number with a units attribute, which should be "bytes", "Kbytes", "Mbytes", "Gbytes" or "Tbytes".
13. controlledVocabulary: A *controlledVocabulary* simply adds an optional vocabulary attribute to the string-valued element, indicating that the value comes from a restricted list.
14. variables: A *variables* element contains a list of variables or a variableMap element that refers to another document that contains a list of variables. This list specifies the variables (aka fields or parameters) that are available in the dataset, and associates them with a standard vocabulary of names, through the vocabulary attribute.

3.1.2 Metadata schema for CS/W

The metadata schema for CS/W is based on the international standard for metadata description ISO 19115:2003/Cor.1:20066. In addition, the encoding of any information object in this profile is based on ISO/TS19139 [1]. The main purpose of the metadata schema is to provide a formal structure for the description of information resources that can be managed by a catalogue service that complies with the application profile [27].

The 19115:2003/Cor.1:2006 specifies a general purpose model for metadata descriptions. The figure 3.1 shows an overview of the basic classes of the metadata schema. The table 3.1 describes the metadata basic classes elements [27, 2].

3.2 CONCEPTUAL FRAMEWORK FOR DATA DISCOVERY

User searching for data in an ideal case would search for these data using a spatial, temporal and attribute subset as mentioned in section 2.1. In this section we look at three possible approaches for providing metadata discovery for input data for hydrodynamic models:

THREDDS ingested into CS/W server with CS/W enabled client: This approach takes the THREDDS catalogue and ingests its metadata into CS/W server at the back end as analyzed by [23]. So that records on flood related metadata from the THREDDS catalogue server are copied into the CS/W server, while the front end (client-side) is a CS/W client which allows for spatio/temporal and attribute subsetting at catalogue level. This allows the user to search for data at a spatial, temporal and attribute scale. One limitation to this approach is that the metadata schema of THREDDS and CS/W are different as described in the previous section and we would need to do a semantic match between metadata schema used in both catalogues. Figure 3.2 below describes this approach.

THREDDS server with custom built client: This framework is made up of the THREDDS server as the only catalogue server, with the design and creation of a custom built user interface and a database to the THREDDS server. This is because from the review made in

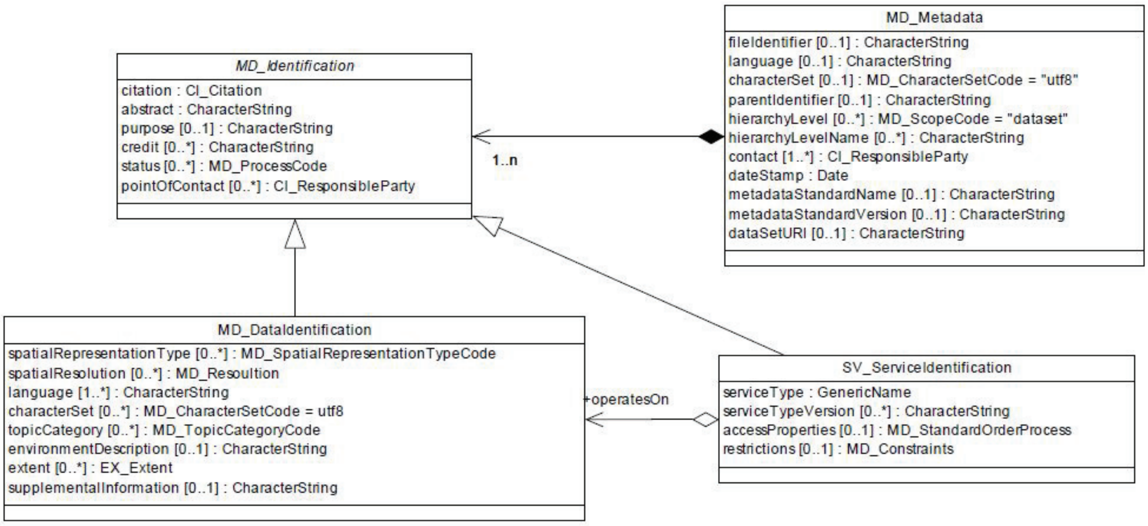


Figure 3.1: Metadata basic classes–Excerpt from 19115:2003/Cor.1:2006

Table 3.1: Metadata basic classes description

Class name	Description
MD_Metadata	Contains Metadata entity set information. The MD_Metadata entity is a composite of MD_Identification and further classes that are suppressed due to clarity, but explained in detail in 19115:2003/Cor.1:2006. [ISO19115:2003 A.2.1]
MD_Identification	This abstract class contains information to uniquely identify the information resource that has to be described. MD_Identification is mandatory. It may be implemented as MD_DataIdentification or SV_ServiceIdentification. [ISO19115:2003 A.2.2]
MD_DataIdentification	Subclass and concretion of the abstract class MD_Identification. According to the application profile, MD_DataIdentification describes either data or applications. [ISO19115:2003 A.2.2]
SV_ServiceIdentification	Subclass and concretion of the abstract class MD_Identification. SV_ServiceIdentification gives a high level description of services according to ISO19119:2005/PDAM 1. A service might be 'loosely coupled' (with no associated data), 'tightly coupled' (with associated data) or 'mixed coupled'. This distinction is done by setting the couplingType attribute of the SV_ServiceIdentification class [see also ISO19119:2005/PDAM 1 7.4.2]

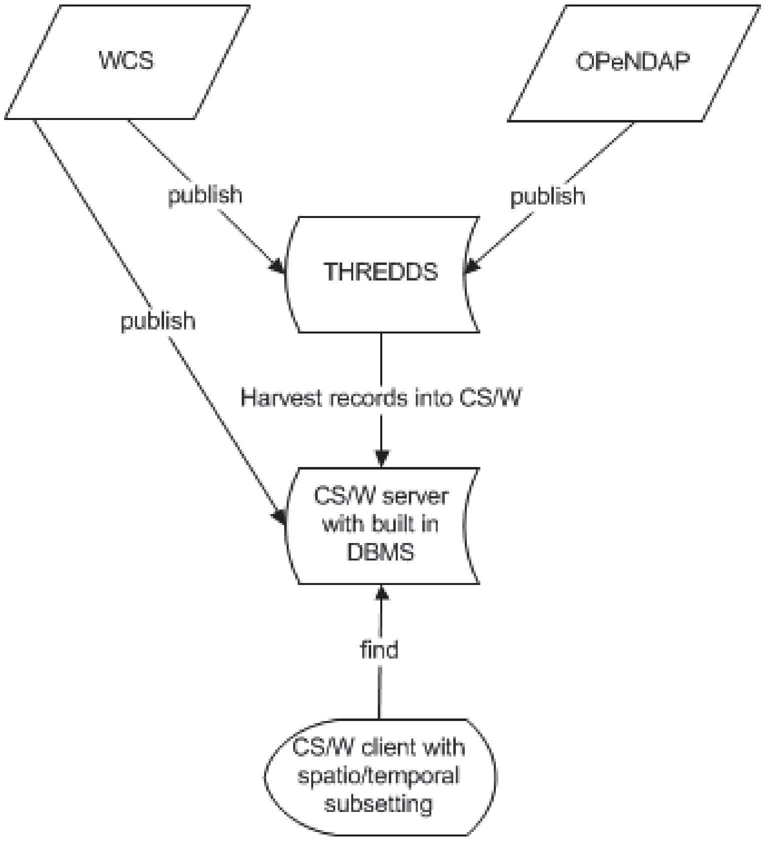


Figure 3.2: THREDDS ingested into CS/W server with CS/W enabled client

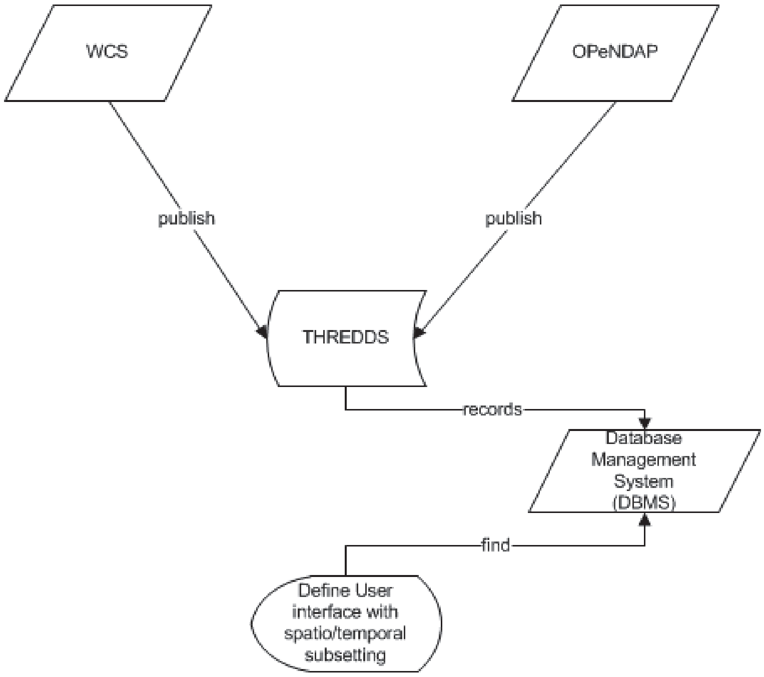


Figure 3.3: CS/W ingested into THREDDS server with custom built client

the previous chapter, most of the data needed as input parameters for hydrodynamic models are found in the THREDDS server. This user interface will be connected to a database at the back end that will store records from the THREDDS catalogue. This is done in order to query the database from the client-side. This newly created user interface should allow for spatial, temporal subsetting and attribute search capabilities. The advantage of this method is that there is no need for metadata schema matching. The disadvantage is that it involves developing from scratch a new user interface with the above functionalities of providing spatio/temporal and attribute subsetting. Another disadvantage is that it involves having to design and implement a database to handle the records from the THREDDS server which is time consuming. Figure 3.3 describes this approach.

THREDDS server with THREDDS enabled client: This framework consists THREDDS server with a THREDDS enabled client on the front end (client-side). The advantage is that there is no need to develop a custom based user interface. But it has its shortcoming which is that THREDDS does not have any client that does spatial and temporal subsetting at catalogue level. THREDDS enabled clients such as Live Access Server (LAS), GraDS Data Server (GDS) and Ingrid only allows for spatio/temporal subsetting at data level but not at catalogue level. So in this case users have to know beforehand which directory to look for data and this does not match our definition of the user and user profile as discussed in the previous chapter. Fig 3.4 gives a schematic description of this framework.

Three approaches for providing a data discovery mechanism are described above. The first best satisfies the objective to the second research question mentioned in chapter 1: How to consistently provide unified metadata access to the end user through CSW and THREDDS? Though from our review most data needed for hydrodynamic models are found on the THREDDS server, this framework can serve as future template for provision of metadata needed for flood extent modeling. This is because it totally encompasses metadata from both CS/W and THREDDS and at the same time can provides spatial, temporal and attribute subsetting.

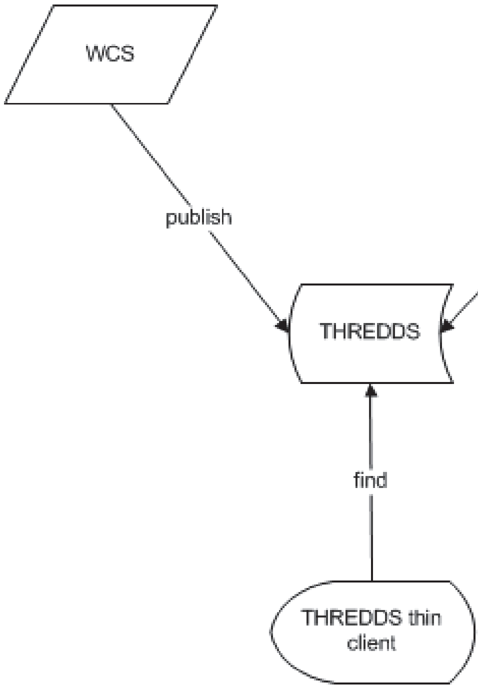


Figure 3.4: CS/W ingested into THREDDS server with THREDDS enabled client

3.3 IMPLEMENTATION OF DATA DISCOVERY MECHANISM

A data discovery mechanism which communicates to the user was implemented, metadata about datasets needed for inundation prediction modeling. This was done based on the conceptual framework described in section 3.2. There was a need to convert metadata format from THREDDS Data Server (TDS) to metadata format compatible with CS/W. For our CS/W server implementation the GeoNetwork opensource version 2.2.0 software which is an OGC compliant catalogue application was used to manage spatially referenced resources this is because it provides powerful metadata editing and search functions.

GeoNetwork uses metadata ISO 19139 profile to describe geographic data and services based on ISO standard 19115:2003 as described in section 3.1.1. This metadata profile is composed of different sections which are [14]:

1. Identification section : This section includes information on citation of the resource (title, date of creation or publication, edition), the abstract, the purpose and the present status of the resource.

2. Distribution section: This section provides metadata elements for accessing other useful on-line resources available through the web.

3. Reference system section: This section defines metadata required to describe the spatial reference system of a dataset.

4. Data quality section: This section provides a general assessment of the quality of the data.

5. Metadata section: This section contains information about the metadata itself.

While the THREDDS is made up of XML files which conforms to the THREDDS Dataset Inventory Catalog Specification 1.0.2. Implementation of the data discovery mechanism, was being run by a Java based interoperable middleware, HarvestThredds Application software. The middleware reaped,transformed and harvested metadata inventories residing in TDS to metadata compatible with GeoNetwork. This application is a software service designed and developed by ITC, Netherlands and UCAR (University Corporation for Atmospheric Research) Unidata Program Center, U.S.A in order to foster interoperability between both GIS and ES communities [49].

The middleware is composed of two layers, the first layer adds to the ability of Unidata TDS to locate datasets on remote data servers published by OPeNDAP servers and then build catalogs and aggregations of data collection on remote hosts. This ability to crawl, catalog and aggregate datasets helps to expose previously undiscovered datasets. The second layer builds on the first layer, it harvests metadata from the TDS including those stored remotely and now catalogable as created from the first layer, in order to augment them with GeoNetwork opensource CS/W server. This augmentation is done by transforming THREDDS XML files complaint to the THREDDS Dataset Inventory Catalog into GeoNetwork ISO19115 XML metadata schema [49].

Transformation from the Unidata TDS XML schema to that of GeoNetwork is done by the HarvestThredds application through the use of an Extensible Stylesheet Language (XSLT), which is a style sheet language for XML documents. XSLT transforms XML document into another XML document. XSLT uses XML Path Language (XPath) query to define parts of the source document that should match one or more predefined templates. When a match is found, XSLT will transform the matching part of the source XML into the new XML.

I customized the HarvestThredds application by constructing a filtering mechanism using keywords such as "srtm", "topography", "bathymetry", "bathy", "radar", "aster", "corine", "modis", "Land cover classes", "avhrr", "landsat", "aster-gdem". These keywords were used as conditions in the filter created in the second layer of the middleware that allowed only dataset inventory necessary for inundation prediction modeling to be converted to metadata suitable for GeoNetwork from the TDS since the transfer of all records from the TDS into GeoNetwork would create a large volume of unnecessary dataset records in the GeoNetwork server. This method created a wider range of discoverable datasets needed for inundation prediction modeling accessible as web services and allowed users to search for using spatial, temporal and subset in order to find datasets. I equally customized the HarvestThredds middleware by enabling it retrieve the dataset URL's from the TDS and harvesting them into GeoNetwork.

Figure 3.5 shows the HarvestThredds application converting metadata from a TDS server to GeoNetwork CS/W profile and harvesting the metadata to GeoNetwork. Two demonstration is shown in figure 3.6 and figure 3.7 of the GeoNetwork user interface showing MODIS land cover class metadata product harvested from the ORNL DAAC TDS (`http://thredds.daac.ornl.gov/thredds/catalog/968/catalog.html`) and bathymetry data harvested from the USGS Woods Hole TDS (`http://geoport.whoi.edu/thredds/bathy_catalog.html`) respectively. In both implementation the middleware converted and copied each dataset node in the THREDDS catalog XML file as a single record into the GeoNetwork server.

3.4 CHALLENGES FOR UNIFIED DATA DISCOVERY

There are a list of constraints that comes with the above framework which is being adopted for unified data discovery for inundation prediction modeling. This is due to the differences between the two metadata schemas which were described in section 3.1.1 and 3.1.2. Thus interoperability can be reached by implementing both structure and syntax mapping from THREDDS to CS/W.

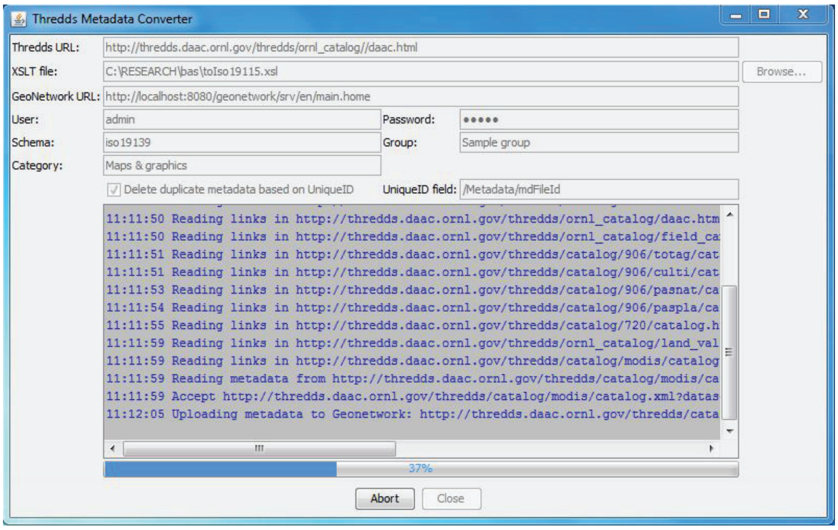


Figure 3.5: Figure of the HarvestThredds application

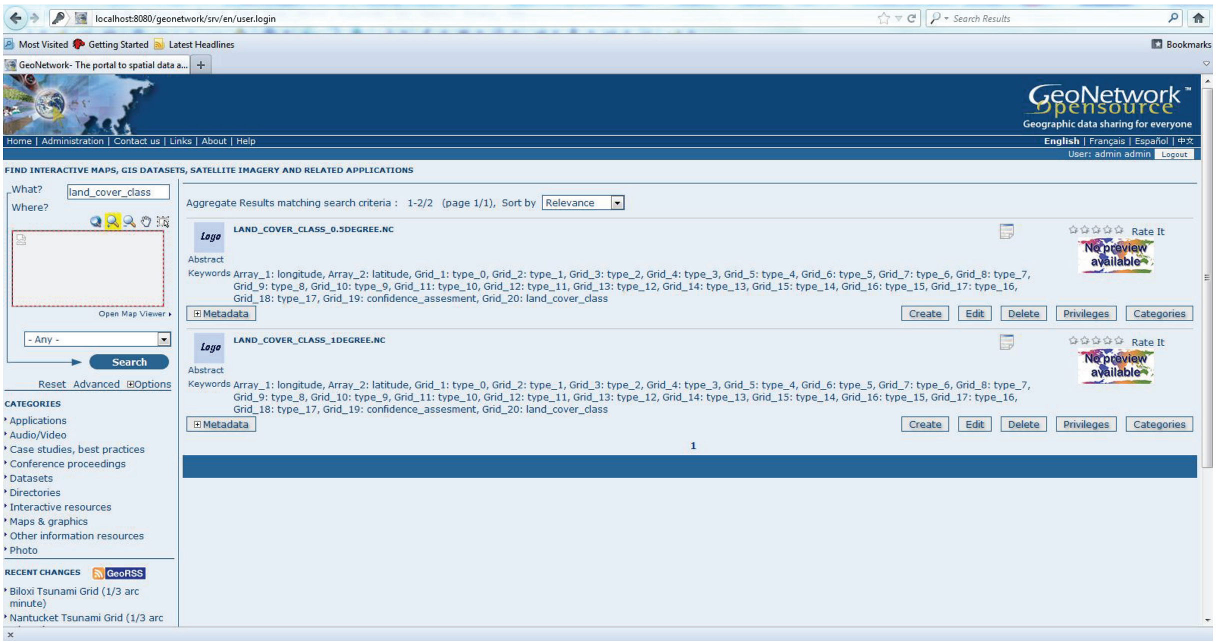


Figure 3.6: Converted metadata of modis product viewed on GeoNetwork interface

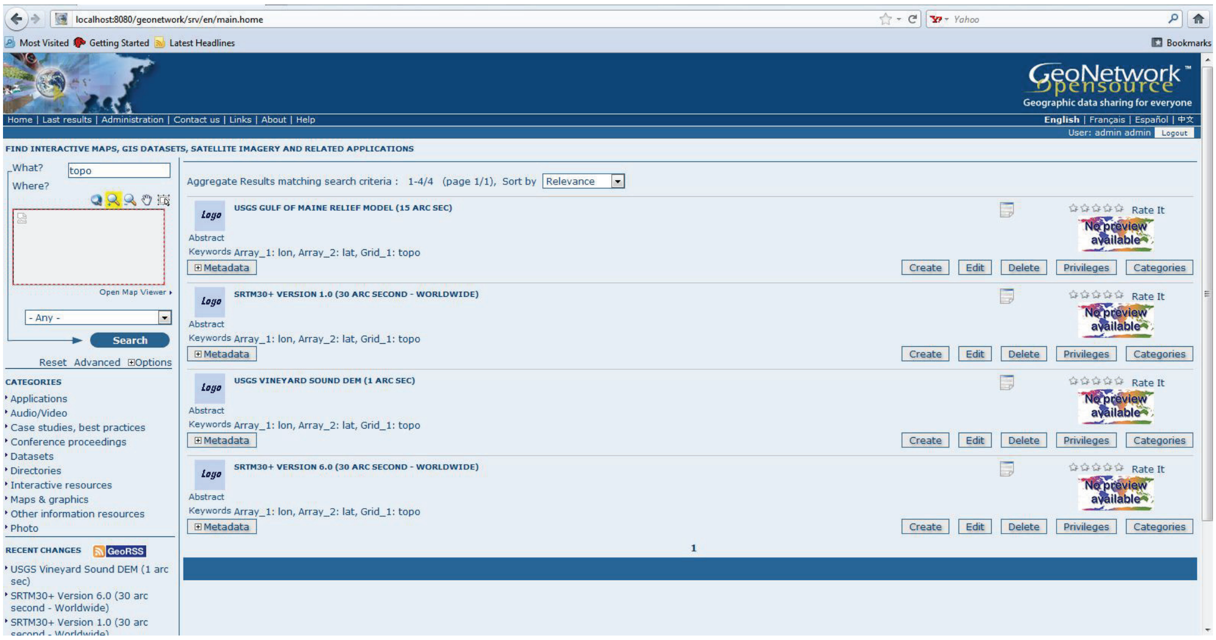


Figure 3.7: Converted metadata of topography viewed on GeoNetwork interface

3.4.1 Structure mapping

For the existence of the approach that was adopted for data discovery, the HarvestThredds middleware structurally matched metadata records from the TDS to that of the GeoNetwork CS/W. THREDDS catalogue uses the hierarchical inventory structure compliant with the THREDDS Dataset Inventory Catalog Specification 1.0.2 as specified in section 3.1.1. This implies that one dataset can include another dataset, in such situation it is known as a parent-child relation and is clearly expressed in the XML file, but this cannot be implemented in the GeoNetwork CS/W profile as analyzed in [18].

The mechanism of the HarvestThredds middleware operates such that: it separates each dataset unit in the THREDDS XML file using XML Path Language (XPath) to select single dataset node as a single unit, so each dataset unit in the THREDDS XML file becomes a single dataset record in the GeoNetwork server. One disadvantage of this approach is that the parent dataset is also transformed into a single GeoNetwork record which is independent of its child dataset record. This results in the child record not inheriting any metadata from its parent record in the GeoNetwork server. Therefore the child dataset does not have a complete metadata since it has lost inherited information from the parent dataset.

An example of a THREDDS bathymetry XML file, part of which is shown in listing A.1. This XML file shows different children bathymetry dataset records within a parent dataset record called bathymetry, the XML file shows the parent dataset with an umbrella metadata from which the children dataset should inherit. But when converted and copied into GeoNetwork dataset records, the parent dataset from the THREDDS XML file becomes a single dataset, and the children dataset does not inherit any metadata from the parent dataset. An easy solution to this problem would be to place a condition that would copy metadata fields automatically from the parent to child, for fields in the child dataset which have missing records.

3.4.2 Syntax mapping

Syntactic mapping means matching elements of the THREDDS and CS/W metadata schema using different expression names but having the same meaning. For example the syntax "name" in THREDDS refers to the same thing as "title" in the CS/W ISO19115 profile which refers the name of the dataset. Likewise, the "//thredds:documentation[@type='summary']" from THREDDS was mapped to MD_Metadata/IdentificationInfo/MD_Identification/citation/abstract, this gives a summary of the dataset. The expression name elements of CS/W information profile was adapted for interoperability. The HarvestThredds application middleware transforms syntactically element names using XSLT. So for each THREDDS XML record and Geonetwork XML record, these different expression syntax that have the same meaning are mapped to each other. A full description of the syntax and element mapping by the XSLT file is shown in table 3.2.

However, some of the elements did not map correctly when implemented by the HarvestThredds middleware, such as in the implemented cases mentioned in section 3.3: When I transferred MODIS land cover class metadata product harvested from the ORNL DAAC TDS (`http://thredds.daac.ornl.gov/thredds/catalog/968/catalog.html`) into GeoNetwork and in the case when the bathymetry records was harvested from the USGS Woods Hole TDS (`http://geoport.whoi.edu/thredds/bathy_catalog.html`) into GeoNetwork.

3.5 CONCLUSION

This chapter described interoperability issues between THREDDS and OGC CS/W with an implementation for common data discovery for datasets needed for inundation prediction modeling and a description of implementation challenges. One of such challenges faced in the implementation was due to the inconsistencies within the structure of the THREDDS catalog XML files for different TDS. Such inconsistency includes metadata information within the parent dataset that is not being inherited from the child dataset when converted and copied into ISO19115 schema for GeoNetwork. This led to incomplete metadata fields in the child dataset. A solution to this problem was proposed in section 3.4.1. Another challenge that was faced in the implementation, was the duplication of records in GeoNetwork by the middleware whenever I converted and transferred metadata from a TDS catalog to GeoNetwork more than once. But this can be solved by adding a function to the middleware that would reject duplicating of already existing datasets in the GeoNetwork server.

Table 3.2: Syntax and element Mapping from TDS (THREDDS Dataset Inventory Catalog) to GeoNetwork (ISO19115:2003)

TDS	GeoNetwork	Definition
/catalog/dataset/ @metadata	/MD_Metadata	this gives general details such as data formats and data types and about the . dataset and the server serving the dataset
/catalog/dataset/ /@name	/MD_Metadata/ IdentificationInfo/ MD_DataIdentification/ citation/CI_Citation/title	Refers to the name of the dataset.
/catalog/dataset/ documentation @type:'summary'	/MD_Metadata/IdentificationInfo/ MD_DataIdentification/abstract	Gives a brief description of the dataset.
/catalog/dataset/ @timeCoverage	/MD_Metadata/IdentificationInfo/ MD_DataIdentification/ extent/EX_Extent/ temporalElememt/ EX_Temporalelement/ extent/TimePeriod	this description of the validity of the dataset
/catalog/dataset/ @geospatialCoverage	/MD_Metadata/IdentificationInfo/ MD_DataIdentification/ extent/EX_Extent/ geographicElement/ EX_GeographicBoundingBox	Defines the geospatial extent of the dataset (Bounding box of entire dataset)
/catalog/dataset/ documentation @type:'summary'	/MD_Metadata/ spatialRepresentationInfo/ MD_GridSpatialRepresentation/ axisDimensionProperties/ resolution/Measure	The summary of the dataset contains a brief resolution of the dataset, this can also be found on the documentation of the dataset in the TDS
/catalog/dataset/ metadata/variables/ @variables	/MD_Metadata/IdentificationInfo/ MD_Identification/ descriptiveKeywords/ MD_keywords/keywords	This gives a list of dataset variables
/catalog/dataset/ /creator@name	/MD_Metadata/IdentificationInfo/ MD_DataIdentification/ pointOfContact/CI_ResponsibleParty/ individualName	This contain name of individual responsible for the dataset
/catalog/dataset/ /date@type	/MD_Metadata/IdentificationInfo/ MD_DataIdentification/ citation/CI_Citation/date	this is the date element but can be presented in different conditions.
/http://<host>/thredds /catalogServices?cmd= subset&catalog= <catalog>&dataset=<ID>	/MD_Metadata/distributionInfo/ MD_Distribution/ transferOptions/online/ CI_OnlineResource/linkage/URL	this contains links to the dataset resource.

Chapter 4

Design and implementation of data acquisition mechanism

Chapter 2 described the users, users profile and user requirements for the prototype web service(s). In addition, WCS, OPeNDAP, THREDDS and CS/W from which we could obtain preprocessed data necessary for input for hydrodynamic models were identified. Chapter 3 gave a detailed description on interoperability issues related to a unified data discovery and the challenges faced. It also described implementation of metadata access to data from both GIS and ES communities. This chapter focuses on the design of a data acquisition mechanism for inundation prediction modeling. In addition with describing challenges faced during implementation of the mechanism.

4.1 CONCEPTUAL FRAMEWORK FOR DATA ACQUISITION MECHANISM

Users would like to retrieve datasets needed for flood extent modeling after they have been discovered. Using the access URL's, users would like to be able to obtain subsets of data in order to reduce the volume of data transferred and to facilitate further processing. These includes being able to obtain spatial, temporal and attribute subsets.

In order to address this requirements, a framework was designed that would be able to retrieve datasets based on the users inputs. It consisted of three basic components which are client, middleware and the server. In the client-side, the user communicates with the system by inputting access URL, bounding box of predefined area, time coverage and attribute variable. The middleware accepts user request, connects to the server and sends request to server. The role of the server which in most cases are remote, is to respond to users request. Figure 4.1 shows the sequence diagram for data acquisition framework.

4.2 IMPLEMENTATION OF DATA ACQUISITION MECHANISM

Data acquisition was implemented and consists of the following components described below:

4.2.1 The client-side

The client side is a stand-alone desktop application developed using TKinter, an opensource Python Graphical User Interface (GUI) package. The user inputs the access URL and bounding box (spatial subset). I adapted the client-side to only accepting access URL's and bounding box only. The interface was used to input the access URL and bounding box is shown in figure 4.2.

4.2.2 Middleware for data acquisition mechanism

The middleware accepts users requests which are access URL's to the discovered datasets from the data discovery mechanism as well as bounding box of predefined area and temporal extents. The

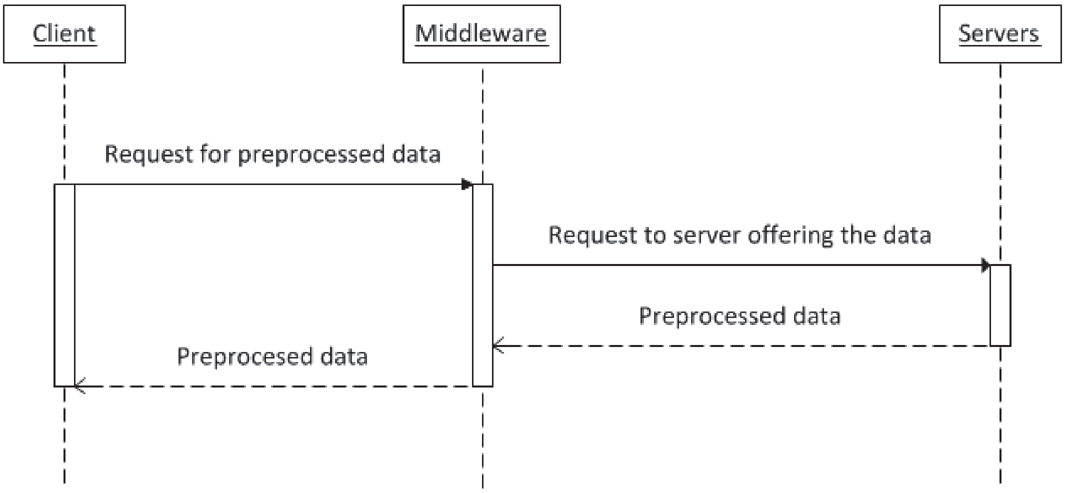


Figure 4.1: Sequence diagram for data acquisition

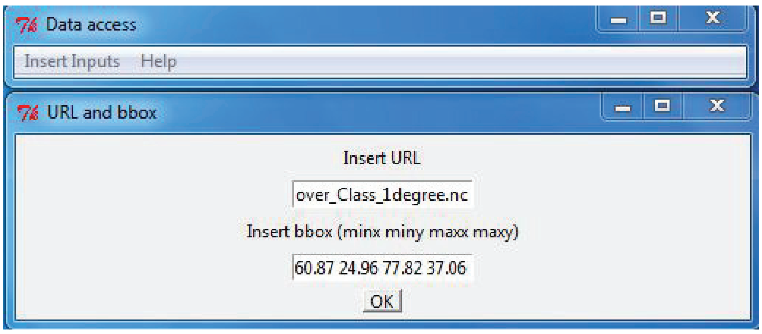


Figure 4.2: User interface for data acquisition mechanism

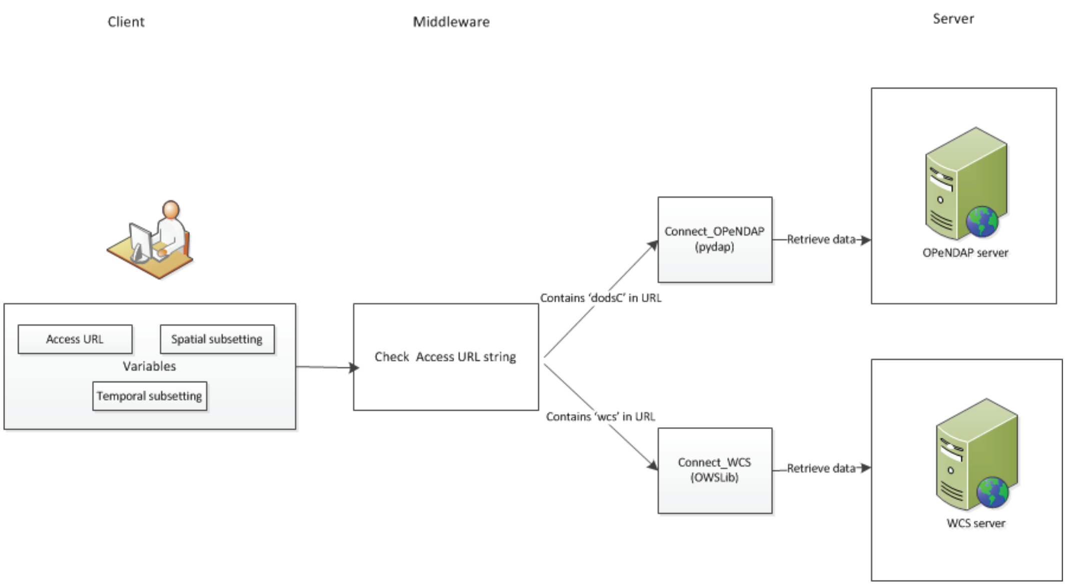


Figure 4.3: Mechansim for data acquisition

middleware then connects to either WCS or OPeNDAP servers depending on the access URL input.

The mechanism of the middleware runs when the user inputs the access URL and the bounding box, a function in the middleware reads the URL and parses the URL as a line of string. If the URL contains 'dodsC' within its string, the middleware then calls a function that automatically connects to the OPeNDAP server and retrieves gridded data inline with the already predefined spatial extent (bounding box). Else if the URL contains 'wcs' within its string, the middleware then calls another function that automatically connects to the WCS server and retrieves gridded data inline with the user's predefined spatial and temporal extent. Figure 4.3 gives a schematic description of the middleware's work flow.

The function used to connect to the OPeNDAP server (connect_OPeNDAP) is a function made up by implementing the Pydap library. Pydap is an open source pure Python library which implements the Data Access Protocol, also known as DODS or OPeNDAP. Pydap can be used as a client to access hundreds of scientific datasets in a transparent and efficient way through the internet; or as a server to easily distribute your data from a variety of formats [41]. It is possible to investigate and manipulate a dataset as if it were stored locally, with data being downloaded on-the-fly as necessary. The Pydap client uses pydap.client.open_url function to open an URL specifying the location of the dataset and to connect to the host OPeNDAP server.

```
>>> from pydap.client import open_url
>>> dataset=open_url('http://geoport.whoi.edu/thredds/dodsC/bathy/
srtm30plus_v1.nc')
```

Pydap can be used to investigate for the variables within the dataset:

```
    >>> print dataset.keys() #check dataset variables
['GDAL_Geographics', 'topo', 'lon', 'lat']
    >>> topo = dataset['topo']
    >>> print type(topo) #check if dataset variable is a grid type
<class 'pydap.model.GridType'>
```

Subsets of data can be obtained with Pydap either by using indexes or by using spatial extents (longitude and latitude).

```
>>> subset_topo = topo[(topo.lat>24.96)&(topo.lat<37.06),(topo.lon>60.87)&
(topo.lon<77.82)]
    return subset_topo
```

The mechanism of the connect_OPeNDAP function works as follows:

- Accept user's access URL.
- Access OPeNDAP server
- Check dataset variables
- Check datatype of dataset variables (if variable is a grid type)
- Retrieve data subset with bounding box (spatial extent), if variable is a grid type

While the function used to connect to the WCS server (connect_WCS) is a function made up by implementing Python module OWSLib. OWSLib is an open source python package for working with OGC map, feature, and coverage services. It provides a common API for accessing service metadata and wrappers for basic WCS operations: GetCapabilities, GetMap, GetCoverage and GetFeature requests [10] and it supports 1.0.0 and 1.1.0 versions of WCS servers. The process of requesting data from a WCS has been described in section 2.3.1. An examples of how to use this OWSLib is to instantiate a WebCoverageService object for a particular WCS service. This will call the GetCapabilities method of the server and populate appropriate python metadata attributes.

```
>>> from owslib.wcs import WebCoverageService
>>> wcs=WebCoverageService('http://geobrain.laits.gmu.edu/cgi-bin/
gbwcs-dem?service=wcs',version='1.0.0')
>>> wcs.identification.title #this is one of the service
 metadata attributes
```

Users can also find out about a particular coverage, such as its spatio-temporal extent, available output formats. This will silently call the DescribeCoverage method on the server to retrieve coverage specific metadata. An illustration on how to investigate the server:

```
>>> wcs.contents.keys()
['SRTM_30m_USA', 'SRTM_90m_Global', 'SRTM30_Plus_Global',
'GLSDEM_90m_Global', 'GTOPO_30arc_Global']
>>> srtm=wcs['SRTM_90m_Global']
>>> srtm.boundingBoxWGS84 #get the spatial extent of the coverage
in latitude longitude
>>> srtm.timelimits #get the temporal extents
>>> srtm.supportedFormats #gets the supported format
```

Using the information gained through a GetCoverage request can be generated and sent to the server and the output of the request can be written to disk or viewed using a software:

```
>>> output=wcs1.getCoverage(identifier='SRTM_90m_Global',
bbox=(-180,-15.0,180.0,61.0),crs='EPSG:4326', format='image/netcdf',
time='', resx='0.01', resy='0.01')
>>> f=open('srtm90.jpeg','wb')
>>> f.write(response.read())
>>> f.close()
```

The OWSLib module was chosen as WCS client because it can be integrated into standard-alone desktop or web-based clients as middleware between software components. And it has the ability to obtain subsets of data.

4.3 CHALLENGES IN IMPLEMENTATION OF DATA ACQUISITION MECHANISM

The implementation of the framework for data acquisition was incomplete due to several challenges. Firstly, both functions were not fully implemented due to time constraint. I retrieved spatial subsets of data but without any implementation for temporal subsets. Temporal subsets where not tested due to the static nature of the datasets that was provided by the web services and was used to implement the mechanism. Secondly, the implementation for the connect_WCS was a static one, that is some of the variables such as data format, projection and resampling parameters used for the GetCoverage request were fixed. This is because in order to make a complete and valid GetCoverage request, there is need to get both GetCapabilities and DescribeCoverage response from the WCS server.

Secondly, TDS with access URL's to service type WCS were not considered in the data acquisition mechanism. This is due to the inability of the TDS to implement complete WCS services such as resampling and reprojection. In the connect_OPeNDAP function, only DAP (DODS) access protocols was considered. While for the connect_WCS, only WCS servers was considered.

4.4 CONCLUSION

This chapter described the conceptual framework and architecture for the partial implementation of a prototype web service that provides the datasets to be used for inundation prediction modeling. Implementation challenges and limitations were described. One such limitation was that temporal subsetting was not tested in this mechanism due to the static nature of the datasets used for implementation. But this can be implemented for dynamic datasets in future implementations since both clients (Pydap and OWSLib) supports temporal subsets. Another limitation was the inability of the TDS to implement complete WCS services such as resampling and reprojection. In addition, limited GetCoverage response data formats (GeoTIFF, GeoTIFFfloat, NetCDF). This restricted the mechanism to retrieve datasets from OPeNDAP protocols only whenever the needed dataset was found on a TDS.

Chapter 5

Design of web service infrastructure

Chapter 3 gave a description and implementation a data discovery mechanism that would allow users discover data that would serve as inputs for inundation prediction modeling. The previous chapter described the design and implementation of a data acquisition mechanism for inundation prediction modeling with a description of its implementation challenges. This chapter gives a description of the design, architecture and software stack for the prototype web service infrastructure that merges both mechanisms above. Thus providing a proof of concept demonstration of a loosely coupled infrastructure for the discovery and retrieval of datasets as well as visualization of such dataset that can be used for inundation prediction modeling.

5.1 CONCEPTUAL FRAMEWORK FOR WEB SERVICE INFRASTRUCTURE

A timely fashion for accessing datasets needed for inundation prediction modeling is required by the user (flood modelers) and users would like to be able to access datasets based on a discovery mechanism at a spatial, temporal and attribute extent. This capability would facilitate inundation prediction modeling and the derivation of flood extent maps from the users.

A conceptual architectural framework for web service infrastructure was designed. This design is made up of both the data discovery mechanism as well as the data acquisition mechanism. This framework consist of three components which are the client, the middleware and the data server (three-tier architecture). Figure 5.1 describes the conceptual design (three-tier architecture) for the web service infrastructure while figure 5.2 shows the sequence diagram for web service infrastructure.

5.2 DESIGN OF SOFTWARE STACK FOR WEB SERVICE INFRASTRUCTURE

The definition of a software stack was based on the user being able to discover and retrieve subsets of data using spatial, temporal and attribute extent.

5.2.1 Choice of clients

This section describes the details for choosing our choice of clients. There are two types of web clients described below which is based on where the data processing is done:

1. Thin Clients: Data processing is done on the server side, the clients serves as user interface to send input request and to view preprocessed output from the server.
2. Thick Clients: Data processing is done on the client side and so performance depends on the client side hardware.

For our prototype web service, thin clients were used due to the processing capabilities of the servers from the data providers. This eluded users the need to have sophisticated softwares installed on their computers in order to have access to data required as inputs for hydrodynamic

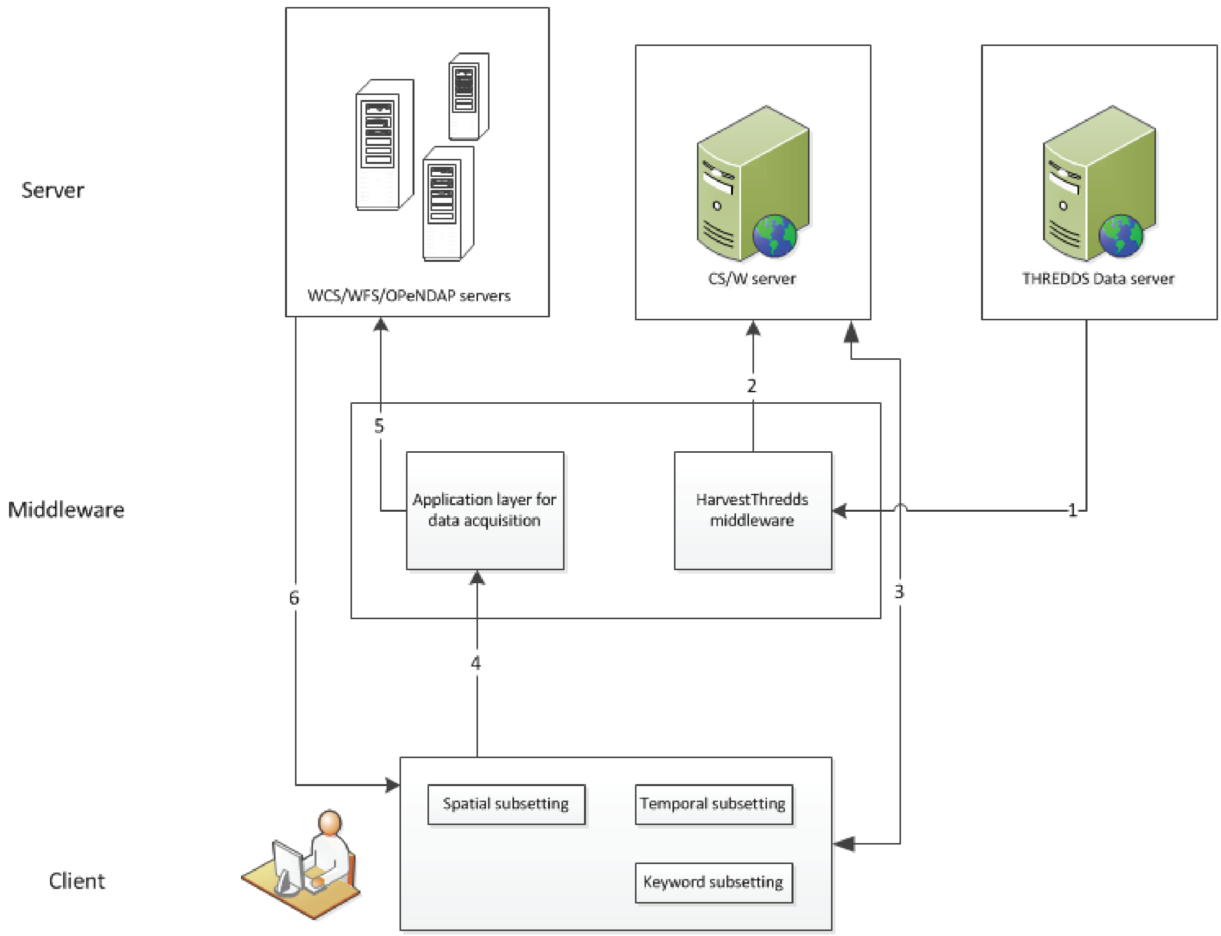


Figure 5.1: Conceptual framework for web service infrastructure

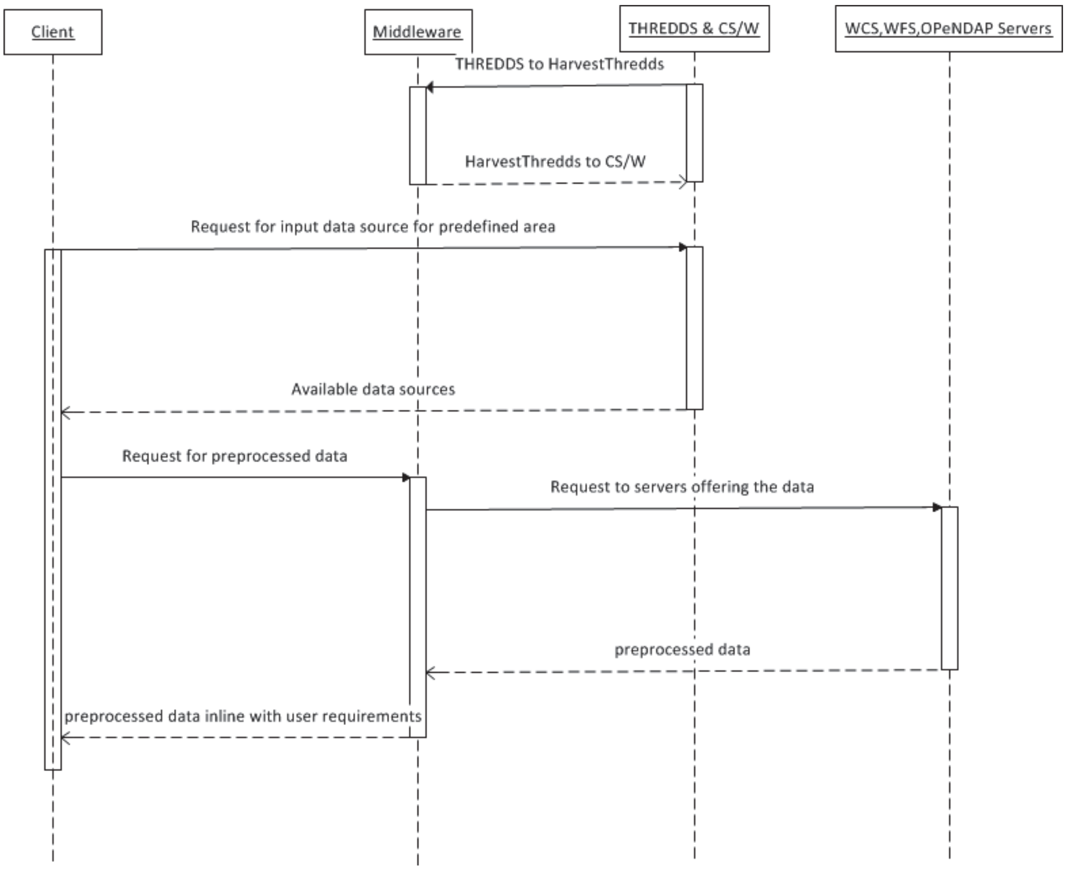


Figure 5.2: Sequence diagram for web service infrastructure

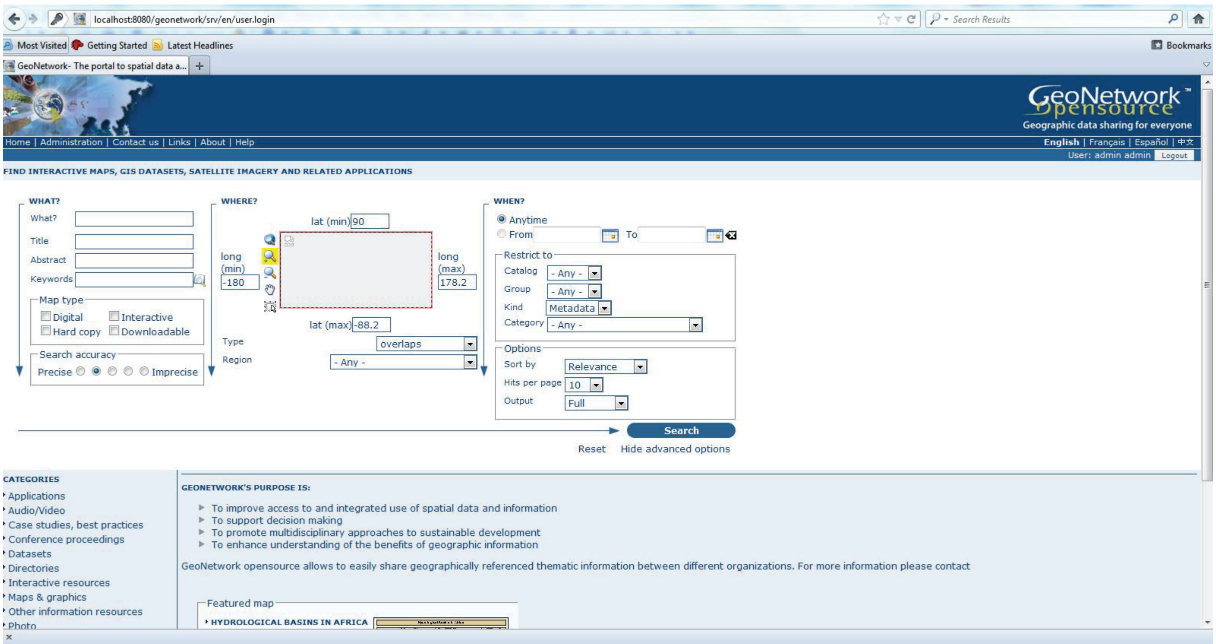


Figure 5.3: Client-side interface

model. Another advantage is that it eluded users of updates made on the server-side. In addition, the performance of the infrastructure was independent of the client hardware.

5.2.2 Client - side user interface

The GeoNetwork CS/W implementation serves as our client since it already provides interface for searching metadata at a catalogue level as well as interactive web map viewer. The interface allows for spatio/temporal subsetting and at the same time allow users to query using keywords. These capabilities of GeoNetwork are shown in figure 5.3. GeoNetwork is stable, flexible, free, it is opensource, platform independent and with a wide community user.

5.2.3 Middleware for web service infrastructure

The middleware for the web service infrastructure is made up of both independent middlewares from the previous two mechanisms. The first middleware is the HarvestThredds application software that would harvest THREDDS datasets into GeoNetwork. While the second middleware layer is the middleware adopted for the data acquisition mechanism. Both middleware for this infrastructure are conceptually designed to adopt the GeoNetwork interface as client–side user interface.

Users query the GeoNetwork for datasets used for inundation prediction modeling after the GeoNetwork server have been populated with THREDDS dataset records. The user then chooses the datasets to be retrieved based on the discovered datasets. The middleware for the data acquisition mechanism now accepts the access URL link, bounding box, temporal scale, and feeds it to the server for response.

Table 5.1: Data discovery mechanism conformance

	Conformance
Functionality	**Data discovery mechanism**
Spatial subsetting	Yes
Temporal subsetting	Yes
Attribute subsetting	Yes

Table 5.2: Data acquisition mechanism conformance

	Conformance	
Functionality	**connect_WCS**	**connect_OPeNDAP**
Spatial subsetting	Yes	Yes
Temporal subsetting	Yes	Not implemented
Attribute subsetting	Yes	Yes
Data resampling	Yes	Not implemented
Data reprojection	Not implemented	Not implemented

5.2.4 Data Servers for web service infrastructure

The servers to be used in this infrastructure consist of the THREDDS and CS/W from the data discovery mechanism for the search of datasets. On the other hand, the infrastructure will also consist of OPeNDAP and WCS from the data acquisition mechanism in order to retrieve discovered datasets as requested by the user that will serve later as inputs for flood extent modeling.

5.3 EVALUATION OF WEB SERVICE INFRASTRUCTURE

Evaluation of the prototype web service infrastructure was done in two parts:

Evaluation of data discovery mechanism: The data discovery mechanism was evaluated based on functionalities described in section 2.1. The functionalities required by the users in discovering datasets were spatial, temporal and attribute extents. Its conformance is shown in table 5.1. The GeoNetwork client–side user interface that was adopted clearly satisfies these basic functionalities.

Evaluation of data acquisition mechanism: The data acquisition mechanism was evaluated based on functionality (basic spatial/temporal/attribute subsetting) and then secondly on performance (speed in which the data is accessed).

A list of functionalities were necessary for users to use this mechanism for effective inundation prediction modeling. The mechanism was evaluated based on the functionalities listed in table 5.2. Table 5.2 evaluates both connect_WCS and connect_OPeNDAP designed in the middleware separately. From the table, processing functionalities such as data resampling and reprojection on remote sensing data such as those needed for inundation prediction modeling were not performed on the connect_OPeNDAP function. This is because the DAP protocol does not support these functionalities.

The prototype web service for data acquisition was equally evaluated based on time consumption in accessing datasets. The first test was conducted with the connect_WCS function for a predefined area (spatial subset). I accessed a spatial subset for SRTM30_Plus_Global (A dataset of world-wide coverage combining NASA's SRTM obtained elevation data and

Smith and Sandwell global 2-minute bathymetry, the dataset provides a global composite of elevation that can be utilized to create elevation/bathymetry visualization) for the country Pakistan with bounding box (60.87, 24.96, 77.82, 37.06) from Geobrain_WCS_DEM (`http://geobrain.laits.gmu.edu/cgi-bin/gbwcs-dem?service=wcs`) which is a WCS server. It took 16 seconds as measured by the python time module to access and retrieve the SRTM30_Plus_Global dataset of size 5.63 Megabytes in netcdf format that was written to disk. This was compared to downloading one–fourth of the entire global SRTM dataset with spatial subset (-45.0, -22.5, 45.0, 22.5) which took 2 minutes and 28 seconds for 111 Megabytes in netcdf format.

The second test was conducted with the connect_OPeNDAP function for the same predefined spatial extent (Pakistan) implemented in the first case. But for MODIS land cover class of 1 degree resolution from the ORNL DAAC's THREDDS Data Server (`http://thredds.daac.ornl.gov/thredds/dodsC/968/Land_Cover_Class_1degree.nc`). It took 40 seconds to print the result to screen. The python time module was used to measure the time taken to print results to screen.

The third experiment was conducted with the connect_OPeNDAP function for the same predefined spatial extent (Pakistan) as implemented in the previous case but for a higher resolution MODIS land cover class product. This case was for MODIS land cover class product of 0.25 degree resolution from the ORNL DAAC's THREDDS Data Server (`http://thredds.daac.ornl.gov/thredds/dodsC/968/Land_Cover_Class_1degree.nc`). It took 59 seconds to print the results to screen. The python time module was used to measure the time taken to print the result to screen.

From the previous two tests, I was unable to write the result to disk using the data acquisition mechanism, due to time constraint and complexity in writing netcdf files from the OPeNDAP server to disk. But alternatively I manually downloaded both MODIS land cover class datasets via the web browser in order to determine the sizes of both datasets. The MODIS land cover class of 1 degree resolution with the above spatial extent was 25 Kilobytes on disk. While the MODIS land cover class of 0.25 degree resolution with the same spatial extent was 219 Kilobytes on disk.

These test were done using a HP personal computer with Intel Core(TM) i5 CPU, 2.53 GHz processor with memory 4.00 GB in 32-bit windows operating system.

5.3.1 Visualization of data residing on the web service infrastructure

This section discusses the visualization of data retrievable from the web service infrastructure. Visualization was done with two different datasets: Firstly, I visualized the MODIS land cover class product of 0.25 degree resolution in netcdf format found on the ORNL DAAC's THREDDS Data Server (`http://thredds.daac.ornl.gov/thredds/dodsC/968/Land_Cover_Class_0.25degree.nc`). This was visualized using ESRI ArcGIS10.0 as client. Two layers (variables) were imported as raster layers from the netcdf file using the multidimensional tools found in ArcToolbox in ArcGIS10.0. The two layers were: the land_cover_class layer as shown in figure 5.4 and the confidence_assessment layer as shown in figure 5.5.

It can be seen from figure 5.4 the different land cover classes for each pixel in the land_cover_class layer. For this particular dataset, the quality of the dataset was communicated in the dataset itself as the confidence_assessment layer. This layer was used to assign quality in terms of percentage confidence for each pixel of the land cover class as shown in figure 5.5.

Secondly, the SRTM30_Plus_Global in geotiff image format for a predefined area (Pakistan) from the Geobrain_WCS_DEM (`http://geobrain.laits.gmu.edu/cgi-bin/gbwcs-dem?service=`

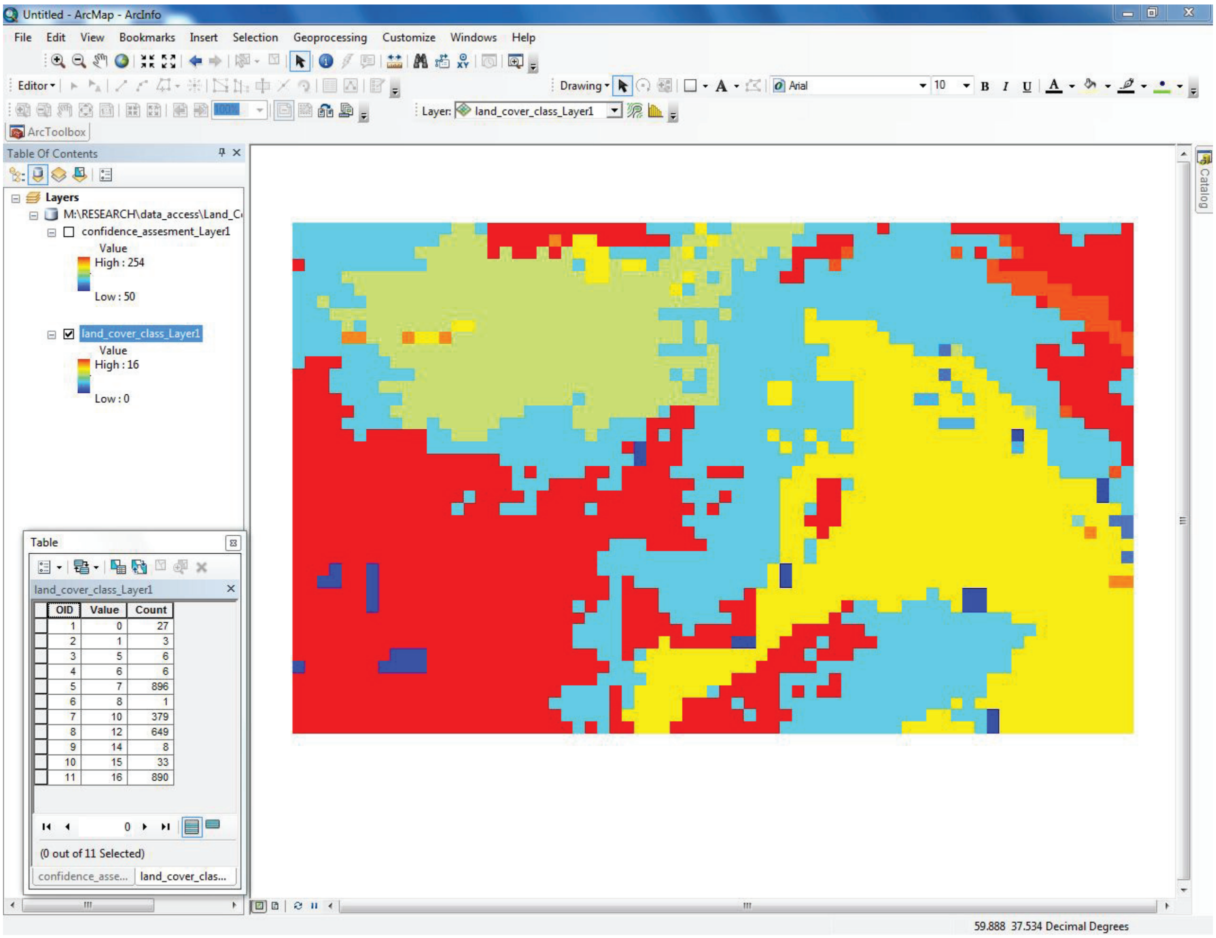


Figure 5.4: Visualization of land cover class layer

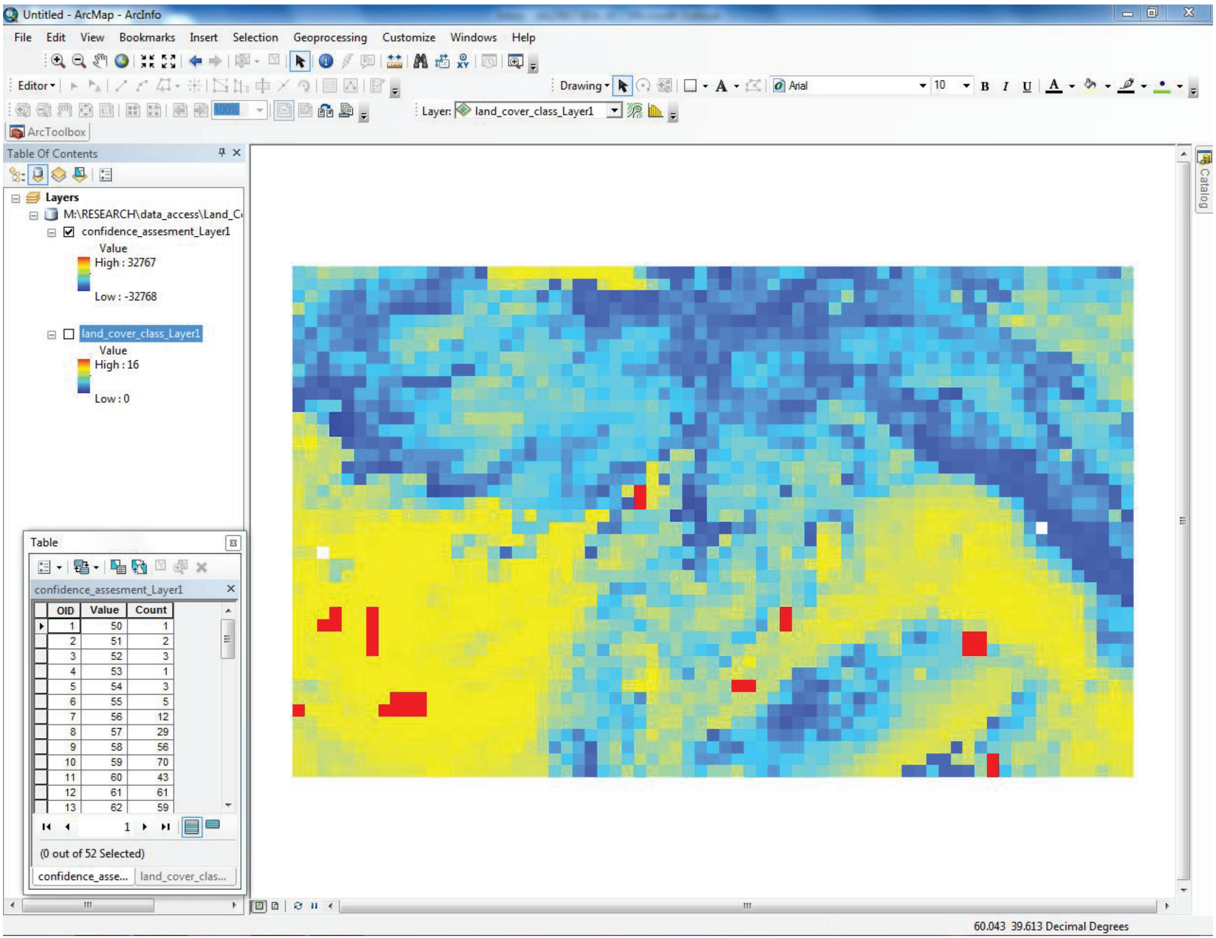


Figure 5.5: Visualization of confidence assessment layer

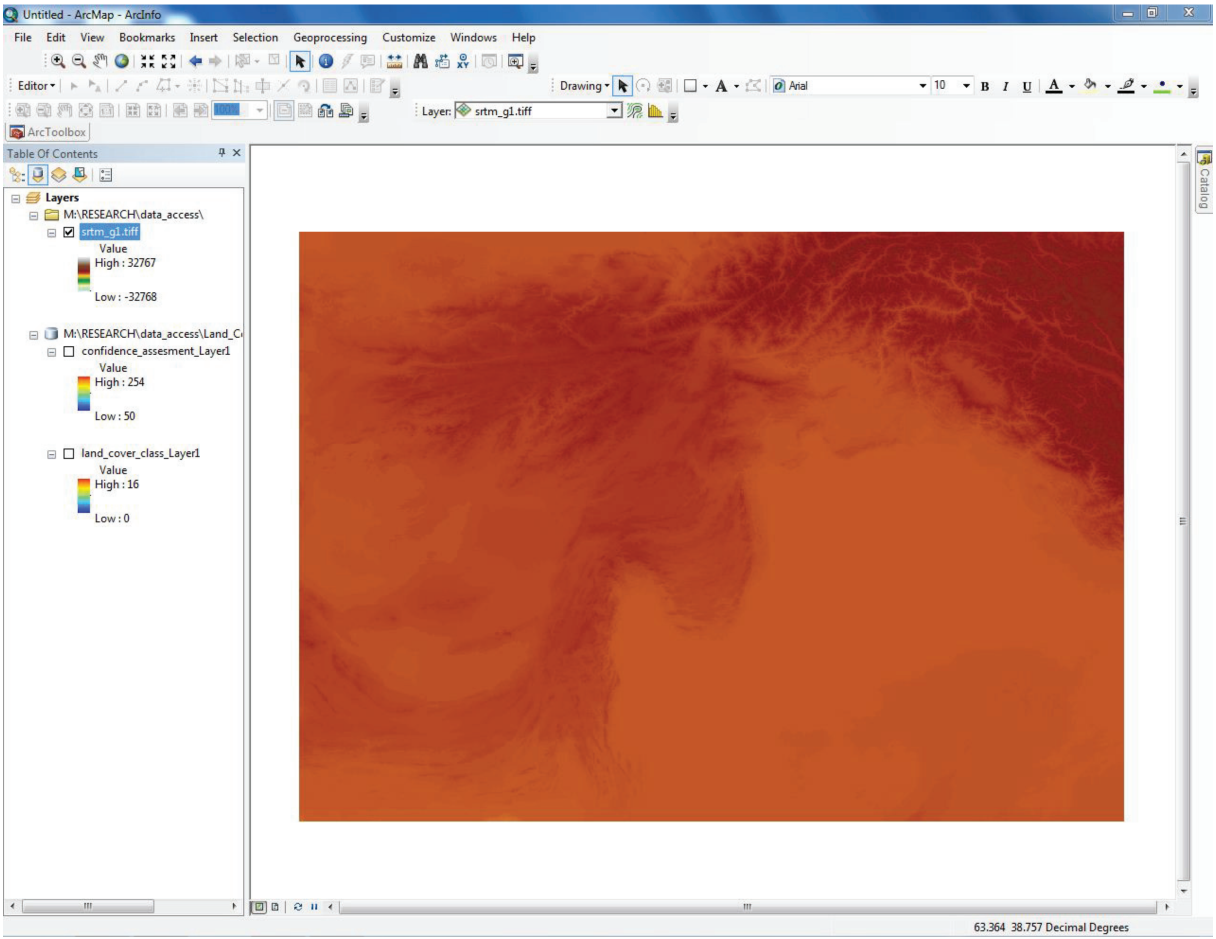


Figure 5.6: Visualization of topographic dataset

wcs) was visualized as well. This dataset has a spatial resolution of 30 arc seconds (approximately 1 kilometer). This was visualized using ESRI ArcGIS10.0 as client. This dataset is shown in figure 5.6.

5.3.2 Applicability of the web service infrastructure for users (flood modelers)

The web service infrastructure is a framework designed to provide easy discovery and access to datasets necessary for flood extent modeling. Easy discovery is based on the large pool of datasets provided by two different catalogue servers from which data choices can be made by the users on type and accuracy of data provided. Easy access is based on retrieving subsets of datasets using either OPeNDAP or WCS protocols. WCS protocols support certain server-side preprocessing functionalities such as resampling and reprojection of gridded data needed by users to facilitate inundation prediction modeling.

The infrastructure made provision for global land cover classes with quality estimates as described in the previous section and 0.25 degree resolution as the finest resolution global dataset on land cover class. This input is needed in hydrodynamic models used to determine friction factor coefficients as a parameter in the hydrodynamic model. The infrastructure equally provides global SRTM datasets with spatial resolution varying from 90 meters to 1 kilometer. This is needed as an input in hydrodynamic model in order to extract the cross-section elevations of rivers and river networks as well as bathymetry of rivers especially in regions where data is scarce. However, the fit for use of these dataset provided is dependent on the user and accuracy of the model employed. Both datasets are global dataset and can be employed in data scarce regions where accurate dataset cannot be found.

The infrastructure did not provide flood extent data (preprocessed ASAR data) without which flood extent modeling cannot be carried out, this was not provided because there currently exist no web service that provides this data using WCS or OPeNDAP protocols. The infrastructure also lacks the provision for line elements (raised roads, embankments, culverts). The presence of line elements improves the accuracy of flood extent modeling especially for very low resolution DEM's as obtained from the infrastructure. This can be provided by WFS services, but this is beyond the scope of the research.

Non remote-sensing data needed as inputs in hydrodynamic models for flood extent modeling such as river bathymetry and boundary conditions (river discharge and water level) are not provided by this infrastructure. This is because the infrastructure takes only remote sensing datasets into account. But these data can be gotten from gauging stations of the area in question.

In all, this infrastructure can serve as base for the provision of all remote sensing datasets needed as inputs for hydrodynamic models for inundation prediction modeling, if these datasets are provided using web service standards (CS/W, THREDDS, WCS and OPeNDAP).

5.4 CONCLUSION

This chapter gave a conceptual description and overview of a prototype web service infrastructure that would be used to discover and retrieve datasets used for inundation prediction modeling. The web service infrastructure was evaluated as a loosely coupled infrastructure, each component (mechanism) evaluated separately. Two datasets residing on the data server of the infrastructure was visualized in order to explore the accuracy and suitability for inundation prediction modeling. The applicability of the infrastructure to users was analyzed. Though no conclusion was reached on its suitability because this is largely dependent on the user, the types of data already acquired by the user and the accuracy of the hydrodynamic model to be used. The conceptual design of this infrastructure is however not devoid of challenges. They are: 1. Incomplete provision of input datasets required for inundation prediction modeling. 2. How to augment both middlewares to use the same user interface both for discovery and acquisition of the data? If the GeoNetwork user interface could save variables (bounding box, time and attribute subset) used for the searching of datasets and later re-use them for the data acquisition mechanism? Instead of the user having to re-input the variables after discovering the data. This could not be solved due to time constraint.

Chapter 6

Results, Discussion, Conclusion and Recommendations

This chapter discusses the results and challenges of implementation. It also answers the research questions posed in chapter 1. Furthermore, the chapter outlined a few areas that are worth looking into that could improve the outcome of this research (recommendation for future research).

6.1 RESULTS

This section states the result of the research and implementation:

Firstly, a literature review was done on users to find out users requirements for datasets needed for inundation prediction modeling, this was described in section 2.1.1.

Secondly, an extensive review on all the free web servers providing these datasets was done, and an inventory of these servers was created. We found two WCS, two CS/W servers, two OPeNDAP servers, six TDS as shown in table 2.3, this review was done by using a THREDDS enabled client (IDV) and a generic web browser (Mozilla) to browse various catalogs as well as searching OGC's complaint product and services.

Thirdly, a data discovery mechanism for searching for these datasets along with their metadata needed by the user for flood extent modeling was designed and implemented. A prototype catalogue (GeoNetwork opensource) server was used for implementation. It also served as a client to view the metadata repository needed for flood extent modeling. The implementation harvested metatdata from two TDS into GeoNetwork. This was done by using a middleware called the HarvestThredds application software that converted metadata from TDS to GeoNetwork CS/W and also harvested these metadata into GeoNetwork, but this middleware was customized to harvest only flood related data into CS/W by creating a filter for records to be harvested. In addition, I customized the middleware by retrieving the dataset URL's from the TDS and harvesting them into GeoNetwork. The results are described in section 3.3.

Fourthly, a data acquisition mechanism was designed and partially implemented in order to retrieve these datasets in respect with users specification from OPeNDAP and WCS servers using python modules Pydap and OWSLib respectively. Limits of implementation were described in section 4.3.

Lastly, a loosely coupled web service infrastructure was designed. This infrastructure was composed from the data discovery mechanism and the data acquisition mechanism. The web service infrastructure was evaluated and the results were described in section 5.3. I visualized a MODIS land cover class product dataset and a DEM (SRTM30_Plus_Global) dataset both in ESRI ArcGIS10.0 for the same spatial extent. The applicability of the loosely coupled infrastructure for users was analyzed based on accuracy and completeness of data provided by the infrastructure for inundation prediction modeling.

6.2 DISCUSSION

There were several challenges which were identified during the research period and are discussed below:

- A review of servers for available data needed for inundation prediction modeling was done. This is because there was no existing inventory for web servers that houses these datasets. An already existing inventory of web servers providing datasets for flood extent modeling would facilitate access to datasets since it enable users to narrow the search for these datasets, based from such an inventory. Although most of the datasets were global datasets. These dataset would need some form of spatial, temporal or attribute subsetting depending on the requirement of the user. It should however be noted that I did not find ASAR data as a web service (in either THREDDS, CS/W, OPeNDAP and WCS) and without these temporally dynamic ASAR data we cannot predict flood extent. In addition, the accuracy of flood inundation prediction is greatly improved in cases with low resolution DEM provided by these web servers, by the provision of line elements (such as raised roads, culverts and embankments) which was not treated within the scope of this research.

- In the design and implementation of the data discovery mechanism, there were three major challenges to the approach that we adopted.

  - The first limitation faced in the implementation was due to the inconsistencies within the structure of the THREDDS catalog XML files for different TDS and the difference in metadata schema between THREDDS and CS/W. Such inconsistency includes metadata information within the parent dataset that is not being inherited from the child dataset when converted and copied into ISO19115 schema of GeoNetwork. This led to incomplete metadata fields in the child dataset. This problem can be solved by to placing a condition that would copy metadata fields automatically from the parent to child, for fields in the child dataset which have missing records. A better solution is to propose same standard metadata schema between the two catalogue servers. If we can smoothly transfer complete metadata information for each dataset records from THREDDS to Geonetwork. Then the usability of such dataset record will depend on the user and the accuracy of hydrodynamic model to be used in the inundation prediction modeling.
  - The second limitation was the difference in syntax adopted by the two catalogue servers for the same meaning. This can be solved by developing a standard naming convention that will be adopted by both catalogue servers.
  - Another implementation challenge that was faced was with middleware (Harvest-Thredds) used, the middleware duplicated records in GeoNetwork whenever I converted and transferred from a TDS catalog more than once. A solution would be to add a function to the middleware that would reject duplicating of already existing datasets in GeoNetwork.

- A mechanism for retrieving datasets was designed, but it was not fully operational due to some limitations discussed in section 4.3. I did not look at the possibilities of other OPeNDAP and WCS clients in accessing these data. This could have further given a better evaluation of the data acquisition mechanism.

- A web service infrastructure was designed. Although this design was not fully operational. A performance evaluation was done on the partially implemented mechanism and the results are discussed in section 5.3. One major shortcoming of this loosely coupled infrastructure was the inability to augment both mechanism into using the same user interface (GeoNetwork user interface). Users had to input spatial, temporal and attribute subsets for each mechanism. This can be solved by augmenting both mechanism to use the same user interface. I could not do this due to time constraint. Another shortcoming of this infrastructure is that it did not provide all remote sensing data needed for inundation prediction modeling. ASAR data (flood extent data) are not provided as a web service and discrete line elements (raised roads, embankments, culverts) are not treated.

- I went a step further to visualize two datasets that was retrievable from the web service infrastructure. The first dataset that was visualized, was the MODIS land cover class product of 0.25 degree resolution in netcdf format for a predefined spatial extent (Pakistan). This was done using ESRI ArcGIS10.0 as client. I viewed two layers from the dataset: 1. The land_cover_class layer and 2. The confidence_assessment layer. The first layer showed the different land cover class for each pixel within the area. While the second layer is used to assess the confidence of each pixel in the first layer. Though the accuracy and completeness of this dataset is dependent on the user and the accuracy of the hydrodynamic model to be used. The second dataset visualized was the SRTM30_Plus_Global (1 kilometer resolution) in geotiff image format for the same predefined area.

- The applicability of the infrastructure for users was analyzed based on accuracy of datasets and the completeness of data for inundation prediction modeling. The infrastructure did not provide a complete set of input dataset for inundation prediction modeling as described in section 5.3.2. Furthermore, the datasets provided presently, were mostly low resolution datasets at a global scale from which subsets could easily be retrieved using certain web service protocols. But for data scarce regions of the world, the fitness for use of these datasets provided would be totally dependent on the user and the hydrodynamic model to be used.

6.3 CONCLUSION

The research was based on providing a consistent infrastructure for providing datasets needed for inundation prediction modeling using web services. The conclusion of this research is summarized by answering the research questions posed in section 1.2.3.

1. How to create an inventory of OPeNDAP and WCS servers which serve data used for inundation prediction modeling?

   Based on users requirements as described in section 2.1.1 an inventory of servers containing datasets necessary for inundation prediction modeling was created as shown in table 2.3. This inventory was created by reviewing these servers using clients of these web servers we used IDV and the general web browser (Mozilla) for browsing through the THREDDS catalogs to determine those THREDDS and OPeNDAP servers. Similarly the list of OGC's complaint services was reviewed to determine which CS/W and WCS servers contained datasets needed for inundation prediction modeling.

2. How to consistently provide unified metadata access to the end user through CS/W and THREDDS?

   Three possible approaches was described to providing a unified metadata access to users in section 3.2, but we implemented the first approach since it best fulfills the users requirements for encompassing all the possible datasets that are available as web services needed for inundation prediction modeling, this combined metadata repository from both CS/W and THREDDS servers and serves as an infrastructure for housing all metadata to datasets available as web services used for inundation prediction modeling.

3. What software stack (Open source or proprietary or a combination) will be used to implement the web service?

   The software stack used for the data discovery mechanism: The GeoNetwork CS/W catalogue server as CS/W server and a CS/W client was used. While for the data acquisition mechanism, I developed a stand-alone middleware application using Python GUI TKinter package and Python modules OWSLib and Pydap to connect and access datasets located in WCS and OPeNDAP servers respectively and retrieve subsets of data. Both mechanism formed the core of the web service infrastructure described in chapter 5 of this thesis.

4. How to determine the quality of data accessed by WCS and OPeNDAP servers based on accuracy and completeness of the data?

   The quality of dataset accessed by WCS and OPeNDAP servers are based on two ways: Firstly, the accuracy of data provided by these protocols was determined in two different cases. In cases where the accuracy in terms of data resolution is communicated in the metadata information as provided by the data providers on these servers. Such as in the global bathymetry/SRTM data provided by USGS Woods Hole TDS (`http://geoport.whoi.edu/thredds/bathy_catalog.xml`) shown in listing A.1 where the accuracy is communicated by the data providers. In cases where the accuracy in terms of confidence of the data can be communicated within the data itself. Such as in the of the MODIS Land cover class dataset product that was visualized in section 5.3.1. Where the confidence of the land cover classes generated from MODIS data was communicated as one of the layers within the dataset itself. The layer in this case was called confidence_assessment layer.

   Secondly, the quality of the data based on completeness of data needed as inputs for inundation prediction modeling was determined. From the study, It was found that there are incomplete datasets provided by WCS and OPeNDAP protocols necessary to carry out inundation prediction modeling. Such datasets include flood extent data and line elements. The absence of boundary conditions and river bathymetry equally makes it impossible to carry out inundation prediction modeling.

5. How to preprocess the data? Which functionalities exist within the above web service standards necessary for preprocessing the data, such as which web standard support resampling of data? If such functionalities exist how it is executed within those standards?

   Server-side preprocessing functionality that existed was resampling and reprojection on coverage data. But these preprocessing functionalities was only implemented on WCS servers alone since OPeNDAP servers does not support either resampling or reprojection of coverage datasets (Land cover and DEM data). It was not implemented on OPeNDAP servers.

6. How to implement the web service(s)?

   The data discovery mechanism was implemented by ingesting records from TDS into GeoNetwork CS/W server using HarvestThredds application which is a middleware application that transforms XML metadata records from TDS to CS/W XML and harvests them into the GeoNetwork server. Furthermore, I implemented a data acquisition mechanism that connected to OPeNDAP and WCS servers to retrieve data based on URL links and Bounding box as inputs by the user. Both implementation are described in sections 3.3 and 4.2 respectively.

7. How to evaluate the implemented web service(s)?

   Evaluation of the prototype web service infrastructure was done based on: 1. Functionality, that is fulfilling all it supposed functions which are being able to discover and search for data using spatial,temporal and attribute extents. 2. Performance (speed in which the data was accessed). 3. The applicability of the web service prototype in terms of providing accurate and complete datasets necessary for inundation prediction modeling.

   The web service infrastructure satisfied the required functionalities, it also showed considerable speed in accessing servers and retrieving subsets of global datasets. But the infrastructure did not show satisfactory results in providing accurate and complete set of datasets required for inundation prediction modeling from which flood extent maps can be created.

6.4 RECOMMENDATIONS FOR FURTHER RESEARCH

The following recommendations are made for future implementation and research:

- More study and implementation should be made on the HarvestThredds middleware. Firstly, to investigate the crawling order of the first layer of the middleware described in section 3.3. Secondly, a more comprehensive structural mapping and syntax mapping between the THREDDS XML and CS/W ISO19115 so that complete metadata information from the THREDDS XML can be transferred smoothly into GeoNetwork for an efficient data discovery mechanism. This means metadata information from higher level directory should be inherited by the child dataset up until the least level directory. This way metadata information is not lost.

- For future experiments between both catalogue servers, I propose standard common metadata schema and naming conventions between the TDS and CS/W servers, this would aid a smoother harvesting of records from either THREDDS to CS/W or the other way round and further aid interoperability between ES and GIS communities.

- A more detailed implementation should be carried out in the the data acquisition mechanism, by making the variables used to make request for data in the middleware dynamic. Furthermore, there is need for more implementation to be carried out with other OPeNDAP clients and WCS client. This should be done in order to further evaluate which clients is best for each of the servers for accessing data used for flood extent modeling.

- Further implementation of the web service infrastructure should done. I propose to research on how both mechanisms can be augmented into using the same user interface (GeoNetwork user interface). This was a major shortcoming of the proof of concept implementation.

- I would propose a data preprocessing mechanism or web service. This should further be integrated into the middleware of the web service infrastructure. This mechanism should preprocess the data gotten from web servers in line with the user requirements before they are used in hydrodynamic models for flood extent modeling. Preprocessing could include resampling and reprojection on DEM data. It should be incorporated in the middleware since preprocessing on the server-side is optional or in some cases not implemented.

- Further study should be made for the implementation of OGC WFS service for the provision of vector data (such as raised roads, culverts and embankments). Prospecting in the use of OpenStreetMaps (OSM) as possible source of line features. This would improve the completeness of low resolution DEM's provided currently by these web servers. Consequently, the accuracy of inundation prediction modeling.

- Investigate the use of GEONETCast (GEONETCast is a Task in the GEO Work Plan and is led by EUMETSAT, the United States, China, and the World Meteorological Organization (WMO) [36]) for the provision of remote sensing products and in situ data, that could serve as data needed as input in hydrodynamic models for flood extent modeling. In addition, exploring the use of EOLI–SA (EOLI–SA is an interactive tool that allows you to access the catalogues of ESA's Earth observation data products, EOLI–SA provides an intuitive way of selecting and ordering Earth Observation data products [11]). This should be done especially for those remote sensing datasets products not treated within the scope of this research, described in section 2.1.1.

LIST OF REFERENCES

[1] ISO/TC 211. Text for ISO/TS 19139 Geographic information - Metadata - XML schema implementation, as sent to ISO for publication. ISO/TC 211 Geographic information/Geomatics, ISO, 2006. [This draft will be checked by ISO before being issued as a Technical Specification-http://www.isotc211.org/protdoc/211n2049/].

[2] ISO/TC 211. Text of ISO 19115:2003 Cor. 1, Geographic information-Metadata-Technical Corrigendum 1. ISO/TC 211 N 1979, ISO, 2006. [http://www.isotc211.org/protdoc/211n1979, 2006-04-11 41pp].

[3] ASTER. ASTER Global Digital Elevation Map Announcement. `http://asterweb.jpl.nasa.gov/gdem.asp`, 2011. [Online; accessed 24-November-2011].

[4] J. Chormanski, T. Van de Voorde, T. De Roeck, O. Batelaan, and F. Canters. Improving distributed runoff prediction in urbanized catchments with remote sensing based estimates of impervious surface cover. *Sensors*, 8(2):910–932, 2008.

[5] P. Cornillon, J. Adams, B.M. Blumenthal, E. Chassignet, E. Davis, S. Hankin, J. Kinter, R. Mendelssohn, J.T. Potemra, A. Srinivasan, and J. Sirott. NVODS and the development of OPeNDAP. *The Oceanography Society*, 22(2), 2009.

[6] P. Cornillon, J. Caron, T. Burk, and D Holloway. Data access interoperability within IOOS. In *OCEANS, 2005. Proceedings of MTS/IEEE*, pages 1790–1792. IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 2005.

[7] USGS LP DAAC. MODIS Overview. `https://lpdaac.usgs.gov/lpdaac/products/modis_overview`, 2011. [Online; accessed 07-October-2011].

[8] G. Di Baldassarre, G. Schumann, and P.D. Bates. A technique for the calibration of hydraulic models using uncertain satellite observations of flood extent. *Journal of Hydrology*, 367(3-4):276 – 282, 2009.

[9] B. Domenico, J. Caron, E. Davis, R Kambic, and S. Nativi. Thematic Real-time Environmental Distributed Data Services (THREDDS): Incorporating Interactive Analysis Tools into NSDL. *Journal for Digital Information*, 2(4), 2002.

[10] B. Domenico and D. Lowe. Using python client libraries to access remote servers via Web Coverage Services. In *25th Conference on International Interactive Information and Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology*, 2009.

[11] European Space Agency (ESA). New EOLI-SA 3.0 Earth Observation data Catalogue. `https://directory.eoportal.org/get_announce.php?an_id=8711`, 2011. [Online; accessed 2-February-2012].

[12] OpenStreetMap Foundation. OpenStreetMap Foundation. `http://www.osmfoundation.org/wiki/Main_Page`, 2011. [Online; accessed 20-October-2011].

[13] J. Gallagher, N. Potter, T. Sgouros, S. Hankin, and G. Flierl. The Data Access Protocol DAP 2.0, 2007-10-10 2007.

[14] Geonetwork. GeoNetwork opensource - The Complete manual. Geonetwork opensource v 2.2, Geonetwork, 2007-2008. [By the Developers,Version 2.2].

[15] K. Gottschalk, S. Graham, H. Kreger, and J. Snell. Introduction to web services architecture. *IBM Systems Journal*, 41(2):170–177, 2002.

[16] A.T. Haile. Integrating Hydrodynamic Models and High Resolution DEM (LiDAR) For Flood Modelling. M.sc thesis, 2005.

[17] A. Hesselink, G. Stelling, J. Kwadijk, and H. Middelkoop. Inundation of a dutch river polder, sensitivity analysis of a physically based inundation model using historic data. *Water Resources Research*, 39(9), 2003.

[18] C.F. Hu, D. Liping, and Y. Wenli. The Research of Interoperability in Spatial Catalogue Service between CSW and THREDDS. In *Geoinformatics, 2009 17th International Conference*, pages 1–5.

[19] P.F. Hudson and R.R. Colditz. Flood delineation in a large and complex alluvial valley, lower Panuco basin, Mexico. *Journal of Hydrology*, 280(1-4):229 – 245, 2003.

[20] M.R. Knebl, Z.L. Yang, K. Hutchison, and D.R. Maidment. Regional scale flood modeling using NEXRAD rainfall, GIS, and HEC-HMS/RAS: a case study for the San Antonio River Basin Summer 2002 storm event. *Journal of Environmental Management*, 75(4):325 – 336.

[21] J. Li and D.W.S. Wong. Effects of DEM sources on hydrologic applications. *Computers Environment and Urban Systems*, 34(3):251–261, 2010.

[22] S. Maharjan. Matching raster and trajectory data using web services. M.sc thesis, 2011.

[23] K. McDonald, Y. Enloe, L. Di, and D. Holloway. A Gateway to Support Interoperability of OPeNDAP and WCS Protocols. In *IEEE INTERNATIONAL GEOSCIENCE AND REMOTE SENSING SYMPOSIUM, VOLS 1-8*, pages 301–304. IEEE, 345 E 47TH ST, NEW YORK, NY 10017 USA, 2006.

[24] Dartmouth Flood Observatory. Space-based Measurement and Modeling of Surface Water. `http://floodobservatory.colorado.edu/`, 2011. [Online; accessed 19-October-2011].

[25] OGC. Web Coverage Service (WCS), Version 1.0.0 (Corrigendum). Technical report, Open Geospatial Consortium, 2005. [OGC 05-076 Version: 1.0.0 (Corrigendum)].

[26] OGC. Web Feature Service Implementation Specification. OGC 04-094 version: 1.1.0, Open Geospatial Consortium, 2005. [OGC 04-094 Version: 1.1.0].

[27] OGC. OpenGIS Catalogue Services Specification 2.0.2 -ISO Metadata Application Profile. 07-045 version 1.0, Open Geospatial Consortium, 2007. [07-045 version 1.0].

[28] OGC. Web Coverage Service (WCS) Implementation Standard. Technical report, Open Geospatial Consortium, 2008. [OGC 07-067r5 version 1.1.2].

[29] OGC. OGC WCS 2.0 Interface Standard - Core. Technical report, Open Geospatial Consortium, 2010. [OGC 09-110r3 version 2.0.0].

[30] OGC. OGC announces netCDF Standard for Communicating Multidimensional Data. `http://www.opengeospatial.org/pressroom/pressreleases/1373`, 2011. [Online; accessed 06-July-2011].

[31] OGC. OGC completes Water Information Concept Development Study. http://www.opengeospatial.org/pressroom/pressreleases, 2011. [Online; accessed 12-August-2011].

[32] OGC. OGC History (detailed). http://www.opengeospatial.org/ogc/historylong, 2011. [Online; accessed 18-August-2011].

[33] OGC. OGC Web Coverage Service. http://www.opengeospatial.org/standards/wcs, 2011. [Online; accessed 28-May-2011].

[34] OGC. OpenGIS Catalogue Services Specification. OGC 07-006r1, Open Geospatial Consortium, 2011. [OGC 07-006r1].

[35] OGC. The OGC's Role in Geosciences and the Environment. http://www.opengeospatial.org/domain/geosciences_and_environment, 2011. [Online; accessed 13-October-2011].

[36] Group on Earth Observations. GEONETCast. http://www.earthobservations.org/geonetcast.shtml, 2011. [Online; accessed 30-January-2012].

[37] OPeNDAP. Available OPeNDAP Client Software. http://www.opendap.org/whatClients, 2011. [Online; accessed 12-October-2011].

[38] OPeNDAP. Dataset List. http://docs.opendap.org/index.php/Dataset_List, 2011. [Online; accessed 29-September-2011].

[39] S.L. Pallickara, S. Pallickara, and M. Zupanski. Towards efficient data search and subsetting of large-scale atmospheric datasets. *Future Generation Computer Systems*, In Press, Corrected Proof, 2011.

[40] S. Patro, C. Chatterjee, R. Singh, and N.S. Raghuwanshi. Hydrodynamic modelling of a large flood-prone river system in india with limited data. *Hydrological Processes*, 23(19):2774–2791, 2009.

[41] Pydap. Pydap. http://www.pydap.org/, 2011. [Online; accessed 27-December-2011].

[42] B. Rabus, M. Eineder, A. Roth, and R. Bamler. The shuttle radar topography mission - a new class of digital elevation models acquired by spaceborne radar. *Isprs Journal of Photogrammetry and Remote Sensing*, 57(4):241–262, 2003.

[43] T. Shengru and M. Abdelguerfi. Web Services for Geographic Information Systems. *Internet Computing, IEEE*, 10(5):13–15, 2006.

[44] W.J. Syme, M.J. Pinell, and J.M. Wicks. Modelling Flood Inundation of Urban Areas in the UK Using 2D / 1D Hydraulic Models. The Institution of Engineers, Australia 8th National Conference on Hydraulics in Water Engineering ANA Hotel Gold Coast, Australia 13-16 July 2004, 2004.

[45] Unidata. Dataset Inventory Catalog Specification, Version 1.0.2 . http://www.unidata.ucar.edu/projects/THREDDS/tech/catalog/v1.0.2/InvCatalogSpec.html, 2009. [Online; accessed 05-November-2011].

[46] Unidata. Overview of Unidata. http://www.unidata.ucar.edu/about/index.html#overview, 2011. [Online; accessed 22-August-2011].

[47] Unidata. THREDDS. http://www.unidata.ucar.edu/projects/THREDDS/, 2011. [Online; accessed 05-August-2011].

[48] unitar. UNITAR OPERATIONAL SATELLITE APPLICATIONS PROGRAMME (UNOSAT). http://www.unitar.org/unosat/, 2011. [Online; accessed 18-October-2011].

[49] V. Venus, A.K. Skidmore, R. Lemmens, A. Toxopeus, B. Retsios, E.R. Davis, C.J.A.M. de Bie, and W. Nieuwenhuis. Harvesting Metadata from THREDDS and OPeNDAP. In *Geophysical Research Abstracts*. Vol. 12, EGU2010-11777, EGU General Assembly 2010, 2010.

[50] Merriam Webster. Merriam Webster. http://www.merriam-webster.com/dictionary/flooding, 2011. [Online; accessed 28-May-2011].

[51] Y. Yamaguchi, A.B. Kahle, H. Tsu, T. Kawakami, and M. Pniel. Overview of Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER). *Geoscience and Remote Sensing, IEEE Transactions on*, 36(4):1062–1071, 1998.

Appendix A

Implementation XML file

Listing A.1: Bathymetry XML file from (http://geoport.whoi.edu/thredds/bathy_catalog.xml)


```xml
<?xml version="1.0" encoding="UTF-8"?>
<catalog xmlns="http://www.unidata.ucar.edu/namespaces/thredds/InvCatalog/v1.0"
    xmlns:xlink="http://www.w3.org/1999/xlink" name="Digital Terrain Models"
    version="1.0.1">
  <service name="allServices" serviceType="Compound" base="">
    <service name="ncdods" serviceType="OPENDAP" base="/thredds/dodsC/"/>
    <service name="wcs" serviceType="WCS" base="/thredds/wcs/"/>
    <service name="ncss" serviceType="NetcdfSubset" base="/thredds/ncss/grid/"/>
    <service name="wms" serviceType="WMS" base="/thredds/wms/"/>
    <service name="fileServer" serviceType="HTTPServer"
    base="/thredds/fileServer/"/>
    <service name="ncml" serviceType="NCML" base="/thredds/ncml/"/>
    <service name="uddc" serviceType="UDDC" base="/thredds/uddc/"/>
    <service name="iso" serviceType="ISO" base="/thredds/iso/"/>
  </service>
  <dataset name="Bathymetry" ID="bathy">
    <metadata inherited="true">
      <serviceName>allServices</serviceName>
      <authority>gov.usgs.er.whsc</authority>
      <dataType>GRID</dataType>
      <dataFormat>NetCDF</dataFormat>
    </metadata>
    <dataset name="USGS Vineyard Sound DEM (1 arc sec)"
    ID="bathy/vs_1sec_20070725.nc" urlPath="bathy/vs_1sec_20070725.nc">
      <documentation
    xlink:href="http://coast-enviro.er.usgs.gov/models/grids/CGSherwo.doc"
    xlink:title="USGS Vineyard Sound Coastal Relief Model (1 arc second)"/>
      <creator>
        <name vocabulary="DIF">WHSC/USGS</name>
        <contact url="http://www.usgs.gov/" email="rsignell@usgs.gov"/>
      </creator>
      <publisher>
        <name vocabulary="DIF">WHSC/USGS</name>
        <contact url="http://www.usgs.gov/" email="rsignell@usgs.gov"/>
      </publisher>
      <geospatialCoverage>
        <northsouth>
          <start>41.0</start>
          <size>1.4</size>
          <units>degrees_north</units>
        </northsouth>
        <eastwest>
          <start>-71.2</start>
          <size>0.5</size>
          <units>degrees_east</units>
        </eastwest>
        <updown>
```

```
      <start>-277.25</start>
      <size>459.6</size>
      <units>meters</units>
    </updown>
  </geospatialCoverage>
</dataset>
<dataset name="SRTM30+ Version 1.0 (30 arc second - Worldwide)"
ID="bathy/srtm30plus_v1.nc" urlPath="bathy/srtm30plus_v1.nc">
  <documentation
xlink:href="http://topex.ucsd.edu/WWW_html/srtm30_plus.html"
xlink:title="SRTM30+ 30 sec Global Topography from UCSD (v1.0)"/>
  <documentation type="Summary">This data consists of 33 files of global
topography in the same format as the SRTM30 products distributed by the
USGS EROS data center. The grid resolution is 30 second which is roughly
one kilometer.</documentation>
  <documentation type="Reference">Becker, J. J., D. T. Sandwell, W. H. F.
Smith, J. Braud, B. Binder, J. Depner, D. Fabre, J. Factor, S. Ingalls,
S-H. Kim, R. Ladner, K. Marks, S. Nelson, A. Pharaoh, G. Sharman, R.
Trimmer, J. vonRosenburg, G. Wallace, P. Weatherall., Global Bathymetry and
Elevation Data at 30 Arc Seconds Resolution: SRTM30_PLUS, revised for
Marine Geodesy, January 20, 2009</documentation>
  <documentation type="Rights">David T. Sandwell, Walter H. F. Smith, and
Joseph J. Becker Copyright 2008. The Regents of the University of
California. All Rights Reserved. Permission to copy, modify and distribute
any part of this gridded bathymetry at 30 second resolution for
educational, research and non-profit purposes, without fee, and without a
written agreement is hereby granted, provided that the above copyright
notice, this paragraph and the following three paragraphs appear in all
copies. Those desiring to incorporate this global bathymetry into
commercial products or use for commercial purposes should contact the
Technology Transfer and Intellectual Property Services, University of
California, San Diego, 9500 Gilman Drive, Mail Code 0910, La Jolla, CA
92093-0910, Ph: (858) 534-5815, FAX: (858) 534-7345,
E-MAIL:invent@ucsd.edu. IN NO EVENT SHALL THE UNIVERSITY OF CALIFORNIA BE
LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR
CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, ARISING OUT OF THE USE OF
THIS global bathymetry, EVEN IF THE UNIVERSITY OF CALIFORNIA HAS BEEN
ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. THE global bathymetry PROVIDED
HEREIN IS ON AN "AS IS" BASIS, AND THE UNIVERSITY OF CALIFORNIA HAS NO
OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR
MODIFICATIONS. THE UNIVERSITY OF CALIFORNIA MAKES NO REPRESENTATIONS AND
EXTENDS NO WARRANTIES OF ANY KIND, EITHER IMPLIED OR EXPRESS, INCLUDING,
BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS
FOR A PARTICULAR PURPOSE, OR THAT THE USE OF THE global bathymetry WILL NOT
INFRINGE ANY PATENT, TRADEMARK OR OTHER RIGHTS.</documentation>
  <creator>
    <name vocabulary="DIF">UCSD</name>
    <contact url="http://www.ucsd.edu" email="jjbecker@ucsd.edu,
dsandwell@ucsd.edu"/>
  </creator>
  <publisher>
    <name vocabulary="DIF">WHSC/USGS</name>
    <contact url="http://www.usgs.gov/" email="rsignell@usgs.gov"/>
  </publisher>
  <geospatialCoverage>
    <northsouth>
      <start>-90.0</start>
      <size>180.0</size>
      <units>degrees_north</units>
    </northsouth>
```

```
        <eastwest>
          <start>−180.0</start>
          <size>360.0</size>
          <units>degrees_east</units>
        </eastwest>
        <updown>
          <start>−10923.0</start>
          <size>19773.0</size>
          <units>meters</units>
        </updown>
      </geospatialCoverage>
    </dataset>
  </dataset>
</catalog>
```