TURNING SPATIAL DATA SEARCH ENGINE TO SPATIAL DATA RECOMMENDATION ENGINE GFM M.SC. RESEARCH

TIGIST SEYUM BESHE
March, 2011

SUPERVISORS:

Ms Dr. I. Ivánová
Dr. J.M. Morales

TURNING SPATIAL DATA SEARCH ENGINE TO SPATIAL DATA RECOMMENDATION ENGINE GFM M.SC. RESEARCH

TIGIST SEYUM BESHE
Enschede, The Netherlands, March, 2011

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: GFM

SUPERVISORS:

Ms Dr. I. Ivánová
Dr. J.M. Morales

THESIS ASSESSMENT BOARD:

Dr.Ir. R.A. de By (chair)
V. de Graaff MSc(External examiner)

ABSTRACT

Search engines are designed to help users to search and find information on the web. The returned search results may contain web pages, images, and other type of files. Hence, finding the right information becomes a trial and error approach resulting difficulty for users. To overcome this difficulty, specialised search engines and recommendation systems are feasible approaches in todays technology. Spatial data search engines are among these specialized search engine dedicated to retrieve geographic information. It allow searching spatial data resources based on spatial extent and location name to return the metadata of spatial datasets. However, the fitness for use decision remain additional task for users. This study contributes towards addressing the rising problem of getting the proper spatial dataset that fits user requirements. The main theme of the study is on understanding users spatial data search quality requirements, designing logic to determine fitness for use of spatial datasets and to implement the fitness for use evaluation logic in a prototype to recommend spatial datasets that best fit user requirements.

The study presents the concept of spatial data quality, fitness for use and recommendation technologies. It further explores the content and structure of metadata of spatial data resources and user spatial data search quality requirements. Overview of spatial data fitness for use evaluation approaches and the recommendation technologies are also well investigated. After these thorough background studies, a data model design for spatial data recommendation profiling and procedures to build profiles is presented. Understanding spatial data fitness for use and recommendation system requires profiling. The spatial data recommendation profiling includes: user profile, spatial data resource profiling, and interaction profiling. Then, the study introduces a new approach and techniques to evaluate fitness of spatial data based on users spatial data search quality requirements. The fitness for use evaluation includes: evaluating spatial data extent with respect to user extent requirement, quantitative and qualitative spatial data quality evaluation based on user specified application and quality requirements.

The proposed fitness for use reasoning logic has been realized through prototype implementation. The research result shows that fitness for use based spatial data recommendation is promising approach to search and recommend spatial data resources based on user specified spatial data search quality requirements. In the proposed scheme the search result based on fitness for use evaluation is enhanced by maintaining spatial data recommendation profile.

Keywords

fitness for use, spatial data recommendation, datasets

ACKNOWLEDGEMENTS

Dear Father God Thank You! At the top most I would like to express my sincere gratitude to my supervisors Ms Dr. I. Ivánová and Dr. J.M. Morales who have supported me through out this thesis work. I am heartily thank you for all the encouragement, guidance and support from the inception to the final level of the thesis work. Without your constructive, important and timely feedback this thesis would not be real. I extend my sincere thanks and gratitude to Ir. V. (Bas) Retsios who supported me during the implementation of my thesis work. I would also like to acknowledge and appreciate all those who contributed in my studies at ITC, particularly the Netherlands Fellowship Programme (NFP) in the provision of scholarship.

My deepest feeling and thanks goes to my dearest husband Mr Fikru Getachew. My dear, your love, support and encouragement is invaluable on both academic and personal level, for which I am extremely grateful. God bless you! I also would like to thank my family who encouraged me, particularly my mother Mrs Yeshi Yemer and my father Mr Seyum Beshe who take care of my son Yegeta Fikru during my study leave. Last but not least, I would like to thank all sisters and brothers in Christ Jesus for all your prayer and encouragement. In my daily work I have been blessed with a friendly and cheerful group of ICF fellowship.

Thank you to all academics and staff at ITC for your guidance, help and continuous cooperation during my study time. God bless you all!

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

DDL Data Definition Language

EA Enterprise Architect

GIS Geographic Information System

IP Interaction Profile

ISO International Organization for Standardization

KML Keyhole Markup Language

PIM Platform Independent Model

PL/pgSQL Procedural Language/PostgreSQL Structured Query Language

PSM Platform Specific Model

SDI Spatial Data Infrastructure

SRID Spatial Reference System Identifier

SDRP Spatial Data Resource Profile

UML Unified Modeling Language

UP User Profile

List of Algorithms

Chapter 1
Introduction

1.1 MOTIVATION AND PROBLEM STATEMENT

Usage of spatial data resources on the web has increasingly become important in daily activates of modern society. This increase in importance is triggered by users' need to access and share spatial data for different purposes such as social, economical and political issues. In web technology mainstream search engines like Google, Yahoo, and Bing are used in accessing distributed information. When Internet users type a keyword or phrase into the search engine query box, they expect a list of search results which can be websites that offer information, products or services related to that keyword. However, finding proper web content is difficult due to availability of vast volume of information on the web. Therefore, searching for proper result requires specialized search engines.

Spatial search engines are specialized search engines primarily dedicated to retrieve geographical information through web technology. They provide capabilities to query metadata records for related spatial data, and link directly to the online content of spatial data themselves. Spatial search engines help users and providers in posting, discovering and exchanging of spatial data. Now a days many spatial search engines are available to support geographical data accessibility (e.g., Geodata [1] , GEO-Portal [2], Metacarta [3], INSPIRE geoportal [4]). They are used to organize content and services such as directories, search tools, community information, and spatial information. Available spatial search engines are based on collaborative process and spatial data resource content standards. The standards ensure consistency among spatial dataset and allow sharing data and integrating multiple sources of information to create an easy to access environment [30].

Spatial data resource providers publish spatial data through search engines to reach spatial data users. Exposing available spatial data to the mainstream search engines is possible through OpenSearch. Moreover, OpenSearch-Geo extensions are developed to facilitate basic geographical data search using Open-Search standard. The main purpose of these extensions is to provide a standard mechanism to query a resource based on geographic extents or location name [49]. OpenSearch-Geo extensions add new parameters of geographic filtering for querying spatial data and recommend set of simple standard responses in geographic format, such as KML, Atom and GeoRSS through spatial search engines [19]. Even though the OpenSearch-Geo extension advance spatial resource search, still decision on fitness for use remains challenging task for users.

The communication method used in spatial search engines is based on standardized Service-Oriented-Architecture [46]. In this service trend catalogue service play significant role. Catalogue service provides a common mechanism to classify, register, describe, search, maintain and access information about resources available on a network [36]. It supports to publish and search collections of descriptive information (metadata) of spatial data resources and related information.

[1] http://www.geodata.gov
[2] http://www.geoportal.org
[3] http://www.metacarta.org
[4] http://www.inspire-geoportal.eu/

Metadata represent resource characteristics that can be queried and presented for evaluation and further processing by both humans and computers.

Metadata is data about data [26] which shows quality and other specifications of the data. Users can get the strength and limitation of spatial data from metadata to understand and filter it based on fitness for use. But the quality descriptions of data that follow predefined quality criteria are not always usable for every user. Users may have different quality requirements towards spatial data resource since their needs depend on their applications.

Spatial data providers and users have different perceptions for quality [28]. The metadata tells about quality of the resource by the resource providers. But the users might have different quality requirement for their application. Besides the users may have a lack of knowledge on interpreting and understanding the metadata. Therefore, they prefer to use the spatial data without any pre-assessment of fitness for use. Sometimes users know only about the application which make use of a spatial data resource. They may not be aware of the requirement of quality assessment over the spatial data resource before use. Due to that often users ignore the quality information and use spatial data that may not best fit their application. This will causes poor decision making and problematic outcomes.

Search is the critical aspect in the path to optimal exploitation of spatial data resources. But, search for spatial dataset remains based on geographical extent and keywords. It means that reasoning for search result is not based on users' quality requirements. Furthermore, current search results are not organized according to users interest. In general in spite of the effort put into making resources available on the web, facilitating the search, and the significant number of spatial data catalogues readily available, it remains a bulky task for users to find out which of the available resources is best fit for use. This is because in the current search engines there is no mechanism supported for users to search and filter resources based on fitness for use.

In this research we propose a search functionality for current spatial data search engines to consider users quality requirements in addition to the geographical extent and keywords matching. This enables search engines to search resources that fit users interest. This indicates the design and implementation of spatial search engines require users quality requirements based communication mechanisms and techniques to use users quality requirements in determining fitness for use of data resources.

Therefore, developing a techniques to determine fitness for use and designing a data model in a way that can be used to recommend spatial data for users based on their requirements is the main focus of this research. Also motivation behind this research work relies on contributing a technique of reasoning logic that can be embedded into current geoportals to enable them to search resources based on fitness for use. Thus, this research aims at enhancing the current geoportals operational functionality and enable various users to get resources according to their quality requirements.

1.2 RESEARCH IDENTIFICATION

The motivation of this research is to develop method for reasoning based on fitness for use to enable spatial search engines recommending spatial data resource for users.

Specific objectives and research questions

1. To review literature on techniques used to decide fitness for use:
   (a) What are the different techniques used to determine user quality requirement?
   (b) What are the different techniques used in recommendation systems?

2. To develop a data model to store information about users, spatial data resources and the interaction between them:
   (a) What is the structure and content of spatial data resource's quality?
   (b) What is the structure and content of user and user requirements for quality?
   (c) What is the structure and content of interaction between user and spatial data resource?
3. To design technique of reasoning to recommend spatial data resource based on fitness for use:
   (a) What are the concepts for reasoning over structured and unstructured information?
   (b) What are the reasoning logics that serve users to get spatial resources, that best fit for their need?
4. To implement the technique in a prototype by using selected scenario:
   (a) How to implement data model about user and spatial data resource and the interaction between them?
   (b) How to implement the recommendation technique that recommend spatial data resource based on reasoning logic?

1.3 INNOVATION AIMED AT

Presently, there is no spatial data search engine that reason out based on consideration of fitness for use. This research has the aim to develop reasoning technique based on fitness for use. It is an innovative idea since there is no similar work.

1.4 METHOD ADOPTED

For the realization of this research, we performed extensive study on the concepts in fitness for use and recommendation technologies. We reviewed literature on fitness for use approach from users and producers perspective in spatial data infrastructure (SDI). We also studied the quality of spatial data and users quality requirements to determine fitness for use. In addition we reviewed the techniques and methods on profiling concept for recommendation system design. As a result we identify the required information to build spatial data recommendation data model based on fitness for use concept.

After we analysed and conceptualized the data model, we designed the profiling algorithm and a reasoning logic to determine fitness for use of spatial data resources. We implement the model and the reasoning logic by using UML modelling language using the Enterprise Architect. For the realization of our system, we use the PostgreSQL spatial database management system with PHP:Hypertext Preprocessor for developing the front end as a web application system. The fitness for use reasoning logic is implemented by using the PostgreSQL structured language(PL/pgSQL) database programming languages.

1.5 THESIS OUTLINE

Chapter one: Introduction

The first chapter of the thesis reviews mainstream search engines, spatial search engines, current achievements on spatial data retrieval on the web and existing problems related to spatial data

search. In addition, the innovation amid at, the research objectives with specific research questions are presented. Finally the methods adopted to address the research objectives are summarized by answering the research questions are summarized.

Chapter two: Fitness for use and recommendation systems

Definitions of fitness for use, spatial data quality description and its relation in determining fitness for use are explained. The fitness for use concepts from spatial data users and producers point of view also reviewed. In addition approaches used to determine fitness for use is presented. Summary of current recommendation technology approaches, and techniques to build profile in recommendation system design are highlighted. Then specific profiling and recommendation techniques chosen for this research are detailed.

Chapter three: Recommendation system data model design and reasoning logic

In this chapter the overview of spatial data recommendation system architecture and conceptual data model for spatial data recommendation based on fitness for use are presented. Also, introduction to spatial data recommendation profiling, detail explanation on how to create spatial data recommendation profile including inputs and its behaviour are given. The profiling procedures and updating system also explained. Then, fitness for use reasoning logic algorithms and description of the algorithm are detailed. In parallel ranking and updating the system following the reasoning logic are presented.

Chapter four: Spatial data recommendation system design

The functional and non functional unit of the proposed recommendation system are explained in this chapter. In addition, the case study selected for prototype implementation is explained in this part of the thesis. The selected application domain in the case study, default quality requirements for application, and spatial data resources description used for a prototype implementation are discussed in detail.

Chapter five: Spatial data recommendation implementation prototype

This chapter focuses on the functionality test to demonstrate spatial data search based on fitness for use reasoning logic. It is achieved by recommending best possible spatial datasets from the system database based on the case study. The system prototype is implemented using PostgreSQL database management system, PL/pgSQL database programming language, PHP: Hypertext preprocessor, JavaScript dynamic web programming language.

Chapter six: Discussion conclusion and recommendation

This chapter discusses conclusions drawn from this research work on use of spatial data recommendation profiling and advantage of fitness for use reasoning logic to recommend datasets. Finally the thesis concludes by recommending future research direction on effective use of the reasoning logic is recommended for further study.

Chapter 2

Fitness for use and recommendation systems: state-of-the-art

2.1 INTRODUCTION

Fitness for use is a way of understanding the relationship between data and the users [24]. The definition of fitness for use has been subjected to the usability of datasets. Among others fitness for use is defined as connector between data quality and the user [48]; meets consumers needs [39]; and user satisfaction [13]. Redman [40] suggested that for dataset to be fit for use it must be accessible, accurate, timely, complete, consistent with other sources, relevant, comprehensive, provide a proper level of detail, be easy to read and easy to interpret. Therefore, fitness for use can be viewed as the capability of the dataset to fit stated user requirements and application specifications.

2.2 DATA QUALITY VERSUS FITNESS FOR USE

Data quality principles are common and universally accepted practice in different fields [9]. Data quality is a perception or an assessment of data's fitness to serve its purpose in a given context and subjective to various applications. It highly depends on the need of individuals on how to use datasets [8]. Quality can be described by object or phenomenon attributes and properties [22]. The term data quality is used to describe the correspondence between an object in reality and its representation in the datasets. Quality can alse be expressed as a measure against a production specification or a user requirements. According to Coote and Rackham [11] there is no absolute high or low quality; quality is relative.

In GIS context the concept of data quality vary and there is no common understanding to have single definition of quality product. A quality product is a product which is free from errors, or a product with confirmation of specifications used, or it can be a product that satisfy users expectations [16]. However, widely accepted expression affirms that spatial data quality is recognized only in terms of its specific use [8]. Mostly the quality definition given by International Standard Organization ISO is accepted in common to describe spatial data quality. The ISO defines quality as the totality of characteristics of a product that bear on its ability to satisfy stated and implied needs[3]. Therefore, for ISO quality is a result that has to be observed during use.

Spatial data quality principles and its characteristics are defined and presented by ISO [3]. The standards mainly describe the spatial data quality using two main categories: quality overview elements and quantitative quality elements. Data quality overview elements provide general, non-quantitative information and are critical for assessing the quality of a dataset for a particular application. These elements include linage, purpose and usage. Quantitative quality elements describe how well a dataset meets the criteria set out in its product specification and provide quantitative quality information [12]. The quantitative spatial data quality elements includes completeness, logical consistency, attributes accuracy, positional and temporal accuracy. These quantitative quality elements are further described by their quality sub-elements. The ISO provide quality

elements with their sub-elements and guidelines for producers to describe the characteristics of the datasets. Spatial data quality evaluation procedure [27] and reporting the result for quality evaluation procedure [26] are also defined by ISO standards.

Devillers and Jeansoulin [16] elaborates the concept of spatial quality by dividing it into two: internal and external quality. Internal quality is used to express the products with no errors. It is associated to express level of similarity that occur between ideal data to be produced and the data actually produced. The internal quality describes characteristics that define the apparent individual nature of products. On the other hand, external quality is used to express products that meet user needs. It is associated to express the similarity between the data produced and user requirements and their needs. Thus, often the definition of external quality overlaps with that of fitness for use.

When data quality description is defined by fitness for use, it should assure the user that the datasets are fit for the intended use. Data quality descriptions must be made to suit the intended use of the data. Good data quality focuses on the most important data, which are critical to the user, and customer driven to satisfy their needs. Generally, fitness for use equates quality with the fulfilment of users specification. The concepts of data quality and fitness for use of spatial data share the characteristics of being dependent on the behaviour of the users and their application. Evaluation of data quality based on user requirement to determine fitness for use requires comparison of the internal and external qualities in a single model [20]. If the spatial data internal quality matches the external quality, then the spatial data is said to be fit for use for the users' intended application.

2.3 FITNESS FOR USE: SPATIAL DATA PRODUCERS' PERSPECTIVE

In Geographic Information Science (GIS) environments spatial datasets frequently have different origins and contain different quality levels. Spatial datasets can be produced using different techniques and processes. In order to determine suitability of a spatial data resource for a certain application, it is necessary to know the inherent characteristics of the resource [22]. The description and quality information of a dataset should explain its characteristics and quality to the data users. As a result, a form of standardization in a way data quality can be described; in order to evaluate the heterogeneous datasets in homogeneous manner, is required as stated by Caprioli [8]. This leads the need to link spatial data resources to a quality specifications as observed from various initiatives.

Currently there are standards that provide a common method to describe, manage, and present the description of spatial datasets and its quality [8]. Therefore, data producers make use of the standard to disseminate the quality description of the dataset. Also success of spatial data providers depend largely on providing information that is fit for the purpose of target users. To achieve this goal understanding the users needs is a key priority. To identify users needs feedback from users are a mechanism used by data producers to improve quality of a dataset.

Producers' perception of spatial data quality mainly depends on the dataset's internal characteristics. These intrinsic characteristics are resulted from production methods, e.g. data acquisition technologies, data models, and storages [45]. Internal quality description of spatial dataset is independent of any task, unless it is collected and processed for a specific application [16]. Producers of spatial data resource assume that users are able to determining a spatial dataset's fitness for use before use of the dataset. Under the fitness for use approach, producers do not make any judgement. They expect users to look at the production quality information and other part of metadata of the spatial dataset and compare it with the list of quality requirements [28]. Spatial data producers provide quality information contents is to help users to determine if spatial

datasets fulfil their application's quality requirements.

To help potential users whether geographical data are fit for the intended use, metadata (data about data) is distributed by data producers [14]. Data producers report what is known about the quality of the data to enable data users to make an informed judgement about the fitness for use of the data. In this approach users are expected to understand the characteristics of a given dataset and the extent of its potential use from metadata [17]. However, allowing users to determine fitness for use by providing data quality information in metadata is one of the failure in GIS.

2.4 FITNESS FOR USE: SPATIAL DATA USERS' PERSPECTIVE

Spatial data users are growing increasingly with increasing and varying needs of geographical information. Spatial data users are grouped in their level of skills to manipulate GIS information. Also there is a wide range of spatial data users based on the type of spatial data they use for their application. However, most of the users ignore spatial data quality description provided with the dataset.

Agumya and Hunter [4] categorised users into three groups based on how they respond to spatial data quality (SDQ) in dataset. The first users group is those who establish fitness for use decision prior to using the data. The second group users rather wish to choose the best among several suitable datasets. Contrary to the other groups, the last group of users use data regardless of their suitability, either because they must use it or they choose to ignore SDQ. According to Oort [38] users ignore the SDQ for different reason that fall into educational and/or technical limitation.

Spatial data users quality requirements are rooted in the intended application they want the dataset to be used for. Users usually evaluate fitness for use of data sources to determine the suitability of data for problem solving and decision making and consider the datasets interoperability with other data sources [16]. In addition, users also determine fitness for use according to their multidisciplinary information needs. Other factors, such as compliance to specific needs and availability of rules and quality control also has impact for users to determine fitness for use [8]. Directly or indirectly users of a dataset need to use information about spatial data quality in order to be able to assess the fitness for use of the data in their context [33].

Even though, geospatial data users need to assess how datasets fit their intended use, information describing data quality is typically difficult to access and understand. From the end-user perspective, metadata are typically not expressed in a straightforward language, they are recorded using a complex structure, and lacking explicit links with the data they describe [33]. Because of these difficulties data quality is often neglected by users, leading to risks of misuse [15]. Users chose to ignore spatial data quality and often used decision-theoretical arguments to motivate their choice [38].

Among others misuse can arise due to the abundant availability of spatial data, enhanced access to these data, and growth of non GIS expert users. Moreover, users failed to determine fitness for use before usage of dataset due to constraints including but not limited to lack of tools, theory and poor documentation of SDQ. Also understanding data quality is a complex task in cases where heterogeneous datasets have to be integrated. Furthermore, the current way of describing spatial data quality are not easily understandable in helping users to decide on potentially useful dataset. According to survey on standard metadata usage the usability result was low [20]. Metadata describes the data from the data producer's point of view and did not help the user to make a decision about the suitability of a dataset for an intended task.

SDQ standards are primarily aimed at data producers data quality specifications (metadata) rather than data users assessment of fitness for use [20]. In contrast, end users frequently do not

use metadata and they sometimes consider metadata is not necessary in ordering datasets [14]. The implication is that there is an increasing quality concept gap between those who use the spatial data and those who are best knows about the quality of the spatial data. Therefore, to narrow the gap between spatial data users and data providers common concept of spatial data quality and evaluation techniques need to be established.

In general due to the heterogeneity of spatial data, the various components of SDQ, unstructured user requirements, and the various reporting approaches determining fitness for use is not an easy task [16]. Furthermore most users have difficulty to identify their quality requirement. They mostly know only the application they are working on and only need the spatial dataset for its realization. Thus, a technique is required to understand users behaviour and their specific quality requirements and application to help them either to properly specify their quality requirement or recommend them the best fitting dataset.

2.5 APPROACHES TO DETERMINE FITNESS FOR USE

Determining fitness for use of a data resource is the only method to avoid risks caused by misuse of spatial data. Comprehensive comparison against user quality requirement and detailed quality description of dataset is the main approach to determine fitness for use. In determining fitness for use users quality requirements, quality description of the dataset, the decision and how it will be influenced by quality are required input parameters [21]. Given these information, evaluation of fitness for use can be implemented. For fitness for use evaluation, the user quality requirement and the dataset quality requirement should have the same base point [20]. Otherwise, with the absence of such common agreement on quality of object, fitness for use assessment become much more complicated.

An approach to determine fitness for use of datasets rely on knowledge about an individual's expertise. Therefore, gather information about users and group them according to their behaviour is critical [21]. Each user group has certain requirements and different aspects of usability that have to be considered. The fitness for use decision can be easily determined if users quality requirement is known.

The well known approach in understanding users quality requirements is translating subjective users requirements into an objective technical specification. The possible quality aspects are assessed from users subjective requirement and adapted to their needs. After the identification of user groups and their requirements, the quality demands can be recognized and assessed [29]. Grum et.al [2] proposed method for systematic and programmable procedure to compute a usability value for each combination of a user requirement and a data quality description of a dataset.

As a general approach in this research the reasoning logic design to determine fitness for use of spatial dataset is also based on comparison of user quality requirement (external quality) and the dataset quality description (internal quality).

2.6 RECOMMENDER SYSTEMS

Recommender systems are widely implemented for searching, sorting, classifying, filtering and sharing a vast amount of information available on the web to allow users to find resources that fit their need. All recommender systems take advantage of a particular set of artificial intelligence techniques [34]. Recommender systems represent user preferences for the purpose of suggesting items to the users so that users are directed toward those items that best meet their needs and preferences. A recommender system customizes its responses to a particular user. Instead of direct response to queries, a recommender system is intended to serve as an information agent of

individual users or group of users [6].

2.6.1 Recommendation techniques

Recommendation techniques have a number of possible classifications [34]. However, all recommender systems have three common fundamental components. The first component referred to as background data is the information that the system had before the recommendation process begins. The second component is the information that users must communicate to the system in order to generate a recommendation. It is referred to as input data. The third is the algorithm that combines background information and input data.

Recommendation techniques can be distinguished on the basis of their knowledge sources which can be the knowledge of other users' preferences, ontological or inferential knowledge about the domain, or added by a users themselves [7]. Determining similar users' interest, and reflecting those interests back in the form of appropriate recommendations, are primary functions of a recommender system. The main classification of recommendation techniques are:

- Collaborative filtering: Collaborative recommendation is probably the most familiar, most widely implemented and most mature among existing recommendation technologies. Collaborative recommender systems aggregate ratings or recommendations of objects, recognize commonalities between users on the basis of their ratings, and generate new recommendations based on inter-user comparisons. Griffith et.al [23] conducted a survey on performance of collaborative filtering.

- Content-based: The system generates recommendations from two sources: the features associated with products and the ratings that a user has given them. Content-based recommender systems treat recommendation as a user-specific classification problem and learn a classifier for the user's likes and dislikes based on product features. A content-based recommender learns a profile of the user's interests based on the features present in objects the user has rated [7]. It is item-to-item or user-to-user correlation. The type of user profile derived by a content-based recommender depends on the learning method employed. Decision trees, neural nets, and vector-based representations have all been used. As in the collaborative case, content-based user profiles are long term models and updated as more evidence about user preferences is observed [23].

- Hybrid recommender systems combine two or more recommendation techniques to gain better performance with fewer of the drawbacks of any individual one. Most commonly, collaborative filtering is combined with some other technique in an attempt to avoid the ramp-up problem [32]. Analyzing the techniques in terms of the data that supports the recommendations and the algorithms that operate on that data, and examines the range of hybridization techniques have been proposed by Burke et.al [7].

Recommender systems typically determine matches via a process of identifying similar users by creating a neighbour users. Determining recommendations based on selected neighbours is named as profile matching [34]. Profile matching involves:

- Find similar users: employing standard similarity measures technique such as Nearest neighbour, Clustering and Classification

- Create a neighbour: techniques used include the creation of centroid, correlation-thresholding, and best-n-neighbours.

- Computing a prediction based on selecting neighbours: some of the techniques used include most-frequent item recommendation, association rule-based recommendation and weighted average of ratings

In this research we take advantage of these three steps of profile matching to enhance the reasoning logic to have best recommendation list.

2.6.2 Profile building and maintenance

The generation and maintenance of accurate user profiles is an essential component of a successful recommender system. Consequently, in analyzing how a recommendation system makes individuals recommendations or assesses a user needs, the key issue is the user profile. A recommender agent cannot begin to function until the user profile has been created.

In recommendation system user profile generation and maintenance require five primary design decisions which are summarized in Table 2.1 as reviewed by Burke et.al [6] and Montaner et.al [34].

2.7 SELECTED RECOMMENDATION TECHNIQUES

After extensive literature review of recommendation systems state-of-the-art the following techniques are selected to build spatial data recommendation.

We choose to use hybrid recommendation approach among recommendation techniques, because it exploit the features of collaborative filtering and content based filtering. The purely content based approach look into the description provided with the item for matching. It lacks subjective data about the items where subjective implies others opinion or usage information about an item. To overcome this limitation the collaborative filtering techniques can be used which provides the subjective data. On the other hand the collaborative system have limitation of an early rating requirement that can be avoided by using content based technique. Therefore, the hybrid techniques enable us to integrate both techniques to achieve reliable recommendation system design.

To build spatial data recommendation system profile, we select "weighted associative networks" approach to represent profiles in our recommendation design. This approach allow us to store users requirements based on their interest and enable us to make fitness for use based comparison. On techniques to generate initial profile we have selected "Manual approach" and "Semi-automatic" techniques. Though users usually are not interested in spending time on establishing their profiles, this is a system requirement to provide satisfactory recommendation results. Therefore it is mandatory for the users to create a profile by registration to use the system. This is how the recommendation system can acquire the minimal profile information to identify users to generate recommendation. In profile learning step the "Not necessary" approach is selected for our system. In user profile information will be learn and used by the system to generate recommendation only when user interact and search resources with explicit input. This technique uses users initial profile in identifying users.

In updating profile, recommendation systems require a feedback techniques. For our system design Explicit and Implicit feedback techniques are selected. The explicit feedback in our recommendation system is not like providing rating, like or dislike; rather it refers to the system updating users' profile based on their explicit search input during every interaction between users and the system. It also refers to the search query modification that can be made by users. In Implicit techniques the system monitor users actions on the data picking. It could also imply the process of finding similar user requirements and dataset usage with in the system. Moreover,

Table 2.1: Techniques required to build and maintain profile

Profile Representation and Maintenance	Techniques	Description
Technique used to represent profile	Weighted associative networks	based on terms and concepts in which user is interested
	Classifier-based models	based on user profiling learning technique; utilizes training sets
	Matrix of ratings and a set of demographic feature	
	Vector space model	represent each item as a vector in a vector space, allowing items with similar content to be assigned similar vectors
Technique used to generate the initial profile	Empty	profile built through recognition of interactions (history-based model)
	Manual	user required to list/register interests
	Stereotyping	user required to complete form containing demographic data
	Training set	user required to rate examples indicating interest
Profile learning technique	Not necessary	system has already acquired information from user registration process
	Structured information retrieval technique	typically, term-frequency or inverse document frequency (TF-IDF)
	Clustering	Similar users are grouped; system assumes members of a group share interests
	Classifiers	Automated classification techniques employing machine-learning strategies
Relevance feedback technique	No feedback	system does not automatically update a profile, so no relevance feedback is required. If desired, user must manually update profile.
	Explicit feedback	typically utilized in systems that require users to indicate like or dislike, participate in ratings, or provide text feedback. Advantage: simple system design; disadvantages: user reluctance to participate in requests for feedback.
	Implicit feedback	preferences by monitoring user's actions, including links followed, click paths, navigation history, and processing actions such as saving, .
	Hybrid approach	combination of explicit and implicit feedback techniques.
Profile adoption technique	Manual	user required to update list of interests
	Add new information	based on relevance feedback technique; disadvantages include inability to delete outdated interests
	Gradual forgetting function	recent user feedback preserved and resulting gradual forgetting of earlier interactions

the implicit techniques refers the system functions to extract spatial data resources required informations from internet and populate in the recommendation data model. Profile adoption is the essential recommendation design element as users' interests change over time. Gradual forgetting function can be used to apply the profile adoption. This technique determine the ability for the system to adapt and reflect recent profiles.

2.8 SUMMARY

In this chapter we build summary of the concepts on fitness for use, spatial data quality, users and producers perspective on fitness for use. Moreover, we summarized the techniques and methods involved in current recommendation system technology. In GIS determining fitness for use is the process of evaluation of data quality with respect to user quality requirement. But it is identified that the perception of users and producers in evaluating fitness for use is different. Quality for user is according to their requirement and quality for data producer is according to data production specification.

Data producers provide quality information to facilitate the determination of fitness for use. It is aimed to allow users to determine whether the dataset fits for their intended use. However due to the increasing availability of online spatial data, services, and diverse user groups, there is high risk in misusing the data. This is worsen with the unavailability of appropriate tools to analyze the data quality as well as the poor documentation. Given the spatial data heterogeneity on the web from multiple sources and increased number of users and their requirements, makes fitness for use computation a difficult task. Therefore, intelligent recommendation techniques that search spatial data based on fitness for use are beneficial to recommend resources that fit user application.

Recommender systems represent user preferences for the purpose of suggesting resources on the web based on user profile. A variety of techniques have been proposed for maintaining profile, and techniques to perform recommendation that includes content-based, collaborative and hybrid techniques. The overall taxonomy of recommendation techniques is common but the implementation is application domain specific. In our research the designed recommendation system applies the recommendation taxonomy in a context of evaluating SDQ with respect to user quality requirement to determine fitness for use of spatial data resources.

The proposed engine, i.e. Spatial data recommendation engine, integrates the recommender system technology and the SDI quality principle to have common quality concept between user and provider to evaluate fitness for use and use the assessed value for best search facility. This approach is characterized by the ability to model and learn user spatial data quality preferences and providers quality perception in common ground. Therefore, in the following chapter data model design and reasoning logic behind spatial data recommendation engine will be addressed in detail.

Chapter 3

Recommendation system data model design and reasoning logic

3.1 INTRODUCTION

In chapter 2, we discussed the concepts of fitness for use from users and producers perspective. Both spatial data users and producers agree that fitness for use evaluation of a dataset before its usage reduce risks caused by misuse of spatial data resource. However, the two sides are not in line with the definition of fitness for use. The concept of fitness for use for spatial data users is the dataset that satisfies their need based on their quality requirement. On the other hand producers express fitness for use as the description of quality description of the dataset. They assume users evaluate fitness for use before usage of the dataset for their application. Hence, the assessment and determination of fitness for use of a dataset remain users' responsibility. However fitness for use computation is not an easy task for users.

In addressing users fitness for use computation difficulty, recommendation technologies are contributing advanced role on recommending resources which are related to the user preference. The main approach of recommendation technologies is based on profiling users information to understand their interest.

Adapting such an approach to the spatial data search engine is of great importance to search spatial data based on fitness for use. In this research work we propose a mechanise to store users spatial data search quality requirements and spatial data quality descriptions that can be used in fitness for use evaluation to recommend spatial datasets to users based on their requirements.

3.2 SPATIAL DATA RECOMMENDATION SYSTEM ARCHITECTURE

The proposed recommendation system design involves three main components as shown in figure 3.1. The figure describes the general view of spatial data recommendation system design framework.

- User interface: allows and controls user system interaction. The recommendation service obtains information about users' need through web based user interface. The user interface design considers user groups [31]. For example, expert users group requires detailed quality information to determine if the resource is useful for their task or not. However, non GIS expert users group lacks understanding about detailed quality information. Therefore, the user interface design should support simple way of allowing these users to specify their data quality requirement. Moreover, if the users group is non human users, special web service communication facility like XML/GML standard data format should be maintained.

- Recommendation system: is the main component of the system which controls the overall interaction to provide fitness for use based spatial data recommendation. It consists of different functional units that manage profiles, retrieve information from system profile database

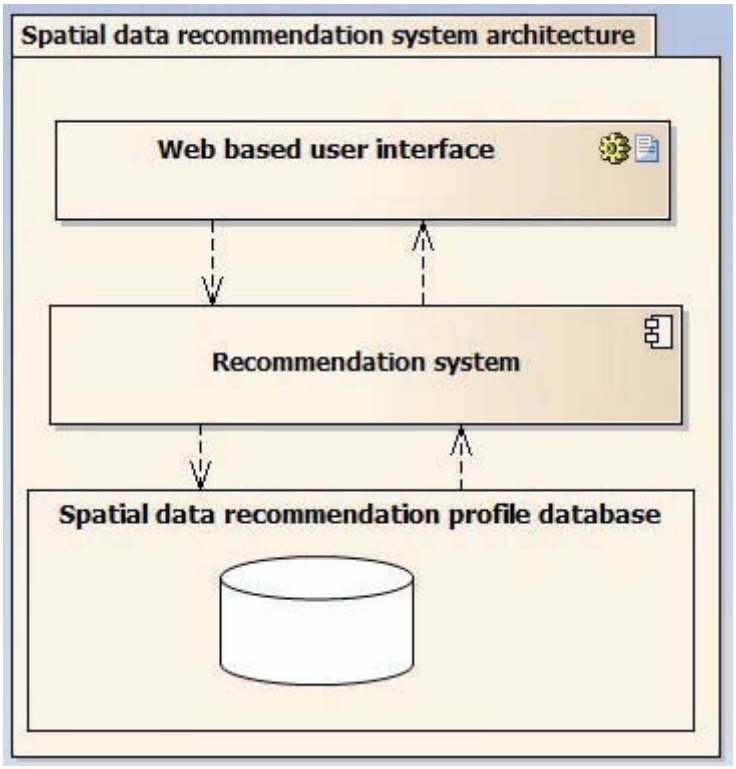

Figure 3.1: Recommendation system architecture

and perform actual recommendation. The functionality of the recommender system should support interaction between spatial data users and spatial data resources. Synchronous interaction within the system should be supported to automate the assessment of fitness for use. The main goal of the system is to recommend spatial data resources to users as per their quality requirement. In achieving this, the system needs to identify user spatial extent, application, and the quality requirements explicitly from user interface or indirectly from users profile.

- Profile database: is the data model of the recommendation system which store users information and spatial data quality information in a structured form. It allows automatic and active data retrieval to speed up the fitness for use evaluation, prediction and recommendation process of spatial data resources. Structured profile storage is defined by the conceptual data model of spatial data recommendation system which is discussed in the following section in detail.

3.3 CONCEPTUAL DATA MODEL OF THE SYSTEM

Well documented conceptual modelling is widely recognized to be the necessary foundation for building a database [42]. It is quite natural that the data model has become the best method to understand and manage information. The concept of data modelling comes from the need for easy access to a structured stored data that can be used for decision making. Without a data model, it would be very difficult to organize the structure and contents of the users requirements and the quality description of the data resources in the spatial data recommendation system as well.

Conceptualizing the data models is especially useful for summarizing and rearranging the data to support the fitness for use evaluation. The goal of the conceptual data model is to represent platform independent information model required by the recommendation system.

The conceptual data model of spatial data recommendation system is designed in a way that allow interactive data retrieval and maintenance during spatial data recommendation process based on fitness for use. The data model given in figure 3.2 describes how user spatial data search quality requirements and spatial data quality description are structured in the system. This static data model of the system supports the fitness for use evaluation and spatial data recommendation process based on users requirements. As shown in figure 3.2, the data model consists of sets of attributes and relationships among different classes for fitness for use based recommendation system. It is used to store users spatial data search quality requirements and the quality description of spatial data resources extracted from catalogue in the web. The, recommender system make use of this static data model to profile required informations and use it in fitness for use evaluation to recommend datasets for users.

After analysing the content and structure of users spatial data search quality requirements and metadata of spatial data resources, the proposed information needed to be profiled in the system. These information includes dataset quality information, spatial extent of the dataset, spatial data resources, users basic information, users quality requirement with weight, and user intended application.

In order to evaluate fitness for use of spatial datasets, in this thesis we followed the spatial data quality principles and quality evaluation procedures as provided by the ISO standard [3], [27]. ISO standard supports common ground about spatial data quality for spatial data users and producers. It outlines the five spatial data quality elements for describing the data quality, and the associated data quality subelements, which we explained in section 2.2. These quantitative subelements quality values are stored in the system data model to be used during fitness for use assessment. However, in order to build quality description of spatial dataset, the specification of spatial data quality requirement details should not be limited to ISO 19113 quality element classification. But any other factors of fitness for use, e.g., usability information, accessibility of dataset, and cost can also be considered [45].

Spatial extent of a dataset describes the geographic area covered by the dataset resources. The extent information allows the system to filter datasets according to the user spatial extent requirement. Therefore, in the conceptual data model we choose to store dataset extent as the geometry polygon. The next section gives detail explanation about information stored in the conceptual data model.

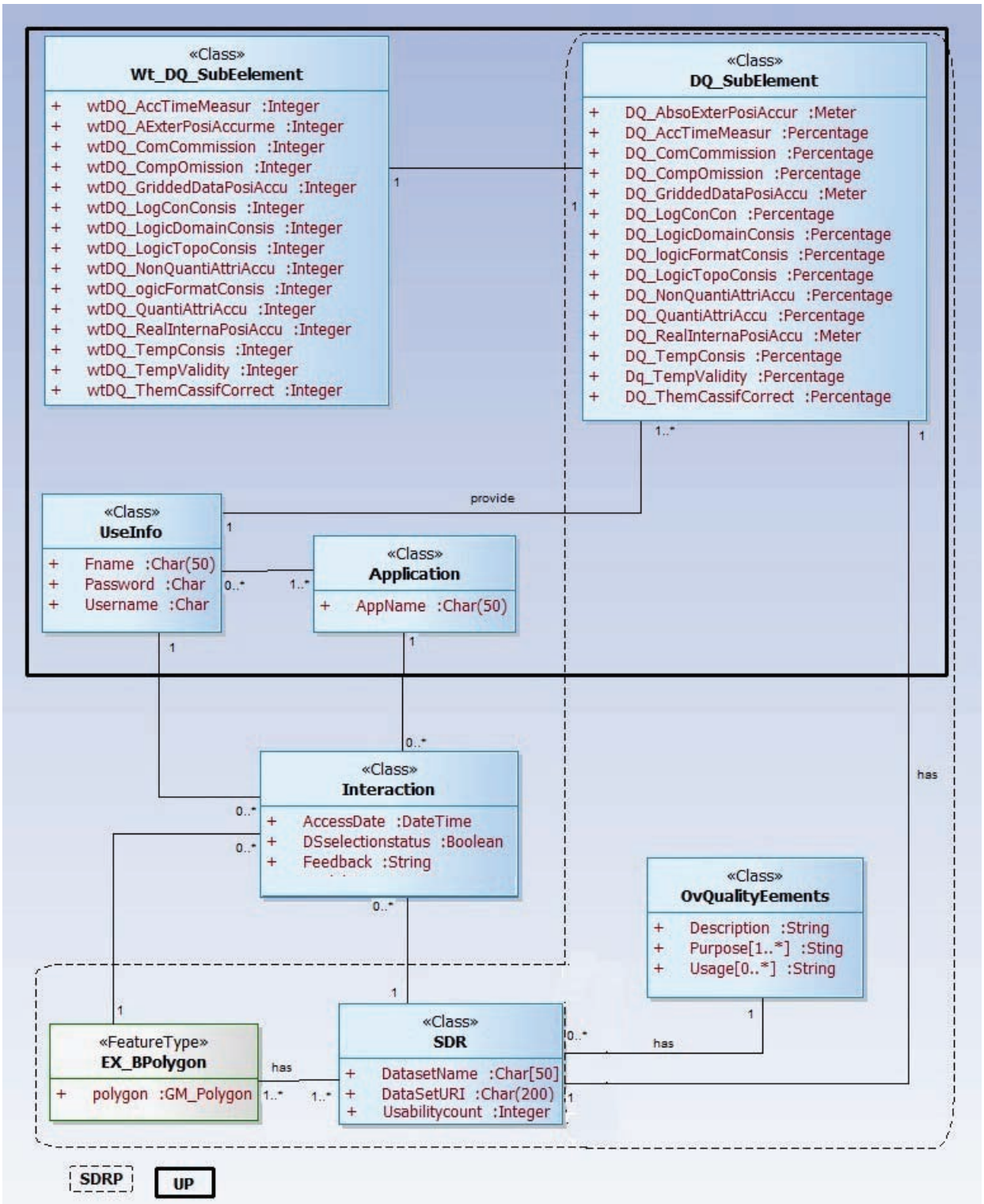

Figure 3.2: Conceptual data modeling of the system

3.4 PROFILING FOR SPATIAL DATA RECOMMENDATION

Profiling is the process of learning users' interests and behaviours. It is used in information retrieval and filtering to provide relevant resource for users. Users information can be collected implicitly or explicitly. Implicit information includes users' behaviour information (e.g., click streams and browsing history), and the content or structural information of the visited web pages or items using some intelligent techniques. Explicit information on the other hand includes users' input information for questionnaire or feedback on the data they used [43]. After collecting the user information, the next step is to analyze the collected information to construct user profiles.

In section 2.6, different profiling techniques were summarized (see table 2.1. Among the techniques explicit feedback from users about their interest and requirement is mentioned as effective way of user profile construction by Montaner [34]. It is because the recommendation system requires the user to provide explicit feedback required in generating reliable recommendation. But, the challenge is that most of the users are not willing to provide feedback [6].

In geographic information retrieval this type of challenge is more difficult. To recommend a spatial dataset that fit users need without knowing user data quality requirement is not possible. Also collecting every possible spatial data users quality requirements explicitly is difficult for reason like users may have limited skills to identify and evaluate their application quality requirements. Therefore, only explicating every profile construction from users or only implicating every profile construction to capture users spatial data quality requirements is not valuable. But combining the two techniques improve the possibilities of getting information to have complete profile. Therefore, in this research, we combined the two methods to build the spatial data recommendation profile.

In the process of spatial data recommendation profiling, based on different information sources and techniques used (explicit or implicit), profiles are grouped into the following sub categories: User Profile (UP), Spatial Data Resource Profile (SDRP) and Interaction Profile (IP). This grouping enable us to distinguish and elaborate the profiling techniques that we have selected in section 2.6.1 and to explain how profiling works in the recommendation system.

To represent spatial data recommendation profile, we considered the "weighted feature vector" approach. In this approach, besides user data quality requirements and spatial data resources quality, the system profile consist weight for each user data quality requirement. Thus, highest weighted quality requirement is used first to determine fitness of spatial data in recommendation. The individual quality element's weight given by users are used for prioritizing and handling the fitness for use evaluation. That means, all spatial data resources are tracked according to the quality requirement evaluation based on quality requirement priority.

Techniques to create initial profile are important to build automated recommendation system [34]. In the spatial data recommendation system creation of initial UP can be achieved using "Manual techniques" which is the realization of explicit method. UP will be initialize at the time they register in the system. Therefore, in this context, explicit profiling means users provide their personal data as well as spatial data quality preferences through user interface. This information will be tracked and stored in the UP.

To initialize SDRP and IP "semi automatic" approach can be used. Semi-automatic techniques work based on combination of explicit and implicit information. Implicit technique is a way of collecting information to build required information in the profile to generate recommendation without the awareness of users. For instance, the recommendation system will make use of user spatial data search requirements to extract data resources from catalogue to build the SDRP, and system builds UP based on default quality requirement per application if users do not provide detail quality requirement.

Updating the existing profile over time is important to build automated and up to date rec-

ommendation system [6]. In spatial data recommendation system updating of profiles performed when users logged into the system and search for data by providing spatial data search requirements. When users use the system and search for spatial data, the system will keep track of users spatial data search requirements and update their profile. Also, the update functionality keeps track of users selected spatial dataset and maintains access information about spatial datasets. Moreover, usability information, which spatial dataset have been used by users, for what application, and with what spatial data quality requirements also maintained.

The outcome of the process of profiling is spatial data recommendation profile which includes structured sets of information about users spatial data search quality requirements and quality description of datasets as provided by the data producers. The spatial data recommendation profile is used to identify users and their requirement, and to determine fitness for use of a spatial dataset based on user requirement.

3.4.1 User profiling (UP)

The proposed user profiling enables the system to maintain information about users' application, their quality requirements, and to identify users to respond with recommendation results according to their requirements. Users' interest changes over time therefore such a system also has to adapt to recent users needs and do the recommendation based on the most recent user's requirement. The UP contains users basic information to identify users and their spatial data search quality requirements. The recommendation system uses these information to search the best spatial data resources that fits users intended use.

To create the profile explicitly, "Structured information retrieval techniques" explained in section 2.7 is used for profile learning. When there is no specific quality requirements explicitly provided by user, the system uses default quality requirements for the users application using "Not necessary techniques" explained in section 2.7 to allow search based on quality evaluation. The default quality requirements definition is given in section 4.5.

The different information representing users profile represented in the conceptual model shown on figure 3.2 are explained as follow:

- User information: to create profile and identify users and their spatial data quality requirements, the system needs to store users basic data with unique login information. The system will make use of this unique login information to identify users and to send response back for their request in a data retrieval session. Also, it is useful in identifying the dataset and application that a particular user have been interested in. In the conceptual model the user basic information is stored in class $UserInfo$.

- User spatial extent requirement: in order to search for spatial data resources, users need to provide spatial extent requirement of their interest. This information will be used by the recommender system to search spatial data resources which have spatial extent matching to the users spatial extent requirement. The users spatial extent requirement stored in the system in a form of polygon geometry.

- Application: users can search spatial data resources by specifying their application that uses the spatial data resources. Using the users intended application, with other quality requirements (if any specified), the recommender system can search for the dataset that fits users' application. That is, though users do not specify quality requirements, the system can make application based fitness for use evaluation to recommend dataset as explained in section 3.5.2. User application information can also used to create usability information about spatial dataset. The application description specified by users is very similar to overview qual-

ity elements defined by ISO 19113 [3]. Therefore, in the fitness for use assessment, users application stored in $Application$ class would be matched with overview quality element about spatial datasets stored in class $OverviewQualityElements$ of conceptual model.

- Users data quality requirements: is basic information need to be specified by the users to make quality based fitness for use evaluation and to generate recommendation. The user quality requirement is stored in $DQ\_Subelements$ class of the conceptual model. The quality evaluation for spatial dataset is according to ISO quality specification as we explained in section 3.3.

- Quality requirement weight: when users provide spatial data quality requirements, they are also expected to provide the weight for each quality element according to their preference. The weights assigned for each quality element are stored in the conceptual data model $Wt\_DQ\_SubElements$ class. The recommendation system make use of these weight to prioritize the elements during fitness for use evaluation process.

The user profiling representation takes users spatial data search requirements including application spatial extent the user interested in, and quality requirement with weight as input. Then it checks the completeness of the quality requirements for the specified application and profile it in the system for spatial data recommendation use. Algorithm 1 gives the high level behaviour of the recommendation system activity in validating users spatial data search requirements in order to represent users information in their profile.

Variable definition used in Algorithm 1 - 2:

- U_E - user extent
- U_A - user application
- $U_Q^i \in U_Q$ - ith user quality requirement where U_Q is set of user quality requirement
- Q_w - set of user quality weight
- S_A - Application name defined in the system
- $S_Q^i \in S_Q$ - ith system application quality where S_Q is set of system application quality
- N_Q - number of quality elements for application
- Q - checked user spatial data search quality requirements

In addition to checking user spatial data search quality requirements based on application, the system checks the weight assigned for user data quality requirements. The algorithm 2 given below shows procedures used in the system to check quality requirements and the assigned weight. If the user does not provide quality value and if the system default quality requirement based on application are used.

Algorithm 1: System checking user spatial data search quality requirement

Procedure:

- For the application provided by the user, check the minimum quality requirement as designed by the system to achieve fitness for use evaluation
- update user quality requirement with system quality requirement if some quality elements are missing in user quality specification
- return checked user spatial data search quality requirements

Input: U_A, U_E, U_Q

1: $S_A \leftarrow$ get_app()
2: **if** $U_A \sim S_A$ **then**
3: **while** $i \leq N_Q$ **do**
4: $U_Q^i \leftarrow$ get_user_quality(U_Q, i)
5: **if** $U_Q^i = NULL$ **then**
6: $S_Q^i \leftarrow$ get_system_quality(S_Q, i)
7: $U_Q^i \leftarrow S_Q^i$
8: **end if**
9: $i \leftarrow i + 1$
10: **end while**
11: **return** $Q \leftarrow \{U_Q, U_E, U_A\}$
12: **else**
13: send_message("**Application not found**")
14: **end if**

Algorithm 2: System checking user spatial data quality elements weight requirement

Procedure:

- Check user quality weight requirement, if some quality elements weight are missing in user quality specification
- return validated spatial data quality weight requirement

Input: U_A, U_E, U_Q, Q_w

1: $S_A \leftarrow$ get_app()
2: **if** $U_A \sim S_A$ **then**
3: **while** $i \leq N_Q$ **do**
4: $U_Q^i \leftarrow$ get_quality(U_Q, i)
5: **if** $U_Q^i \neq NULL$ **then**
6: $Q_w^i \leftarrow$ get_quality_weight(Q_w, i)
7: **if** $Q_w^i = NULL$ **then**
8: send_message("**Insert quality weight**")
9: **end if**
10: **end if**
11: $i \leftarrow i + 1$
12: **end while**
13: **return** $Q \leftarrow \{U_Q, U_E, U_A, Q_w\}$
14: **else**
15: send_message("**Application not found**")
16: **end if**

3.4.2 Spatial data resource profiling (SDRP)

It is generally accepted that spatial data quality descriptions serve users to evaluate fitness of data for their particular application [20]. Different spatial dataset quality model can be used to structure, manage and organize quality information and other required description of the spatial dataset. The data quality model facilitates access to the quality information about datasets to be evaluated and discovered to determine fitness for use [50].

One of the most commonly used spatial data quality model implementations is metadata catalogue which contain description of spatial data including the quality information [36], [37]. A metadata catalogue usually describes location of a dataset and include quality information about the dataset. This is useful for easy discovery and retrieval of spatial data and its quality information [51]. The metadata catalogue service provide efficiency for spatial query search operations. It enables rapid response to detailed data discovery and allows queries and value extraction over metadata attributes [36].

ISO 19139 provides the XML implementation schema for ISO 19115 which specifies the metadata record [44]. In the standard there are only minimum metadata set that are mandatory. Therefore, in reality not all metadata of spatial resources are provided with detail quality description. Moreover, even if the format and structure are based on the standard, the required quality information about the dataset may not be provided by the data producer. Such factors may cause problem in assessment of fitness for use of spatial data resources. In this research, we assumed spatial data resources are provided with metadata description according to the specification of ISO standard including data quality information. We use the quantitative spatial data quality elements for fitness for use assessment based on ISO 19113 quality principle [3].

We need to represent the quality information about the spatial data in a structure that suits the assessment of fitness for use in the system. The spatial data resources profile initialization can be achieved when there is a user spatial data request. This can be done automatically by sending request to metadata catalogue and extracting required quality information. This is where the implicit profiling techniques achieved in the process of building SDRP.

The data model of SDRP based on users spatial data search quality requirements shown in figure 3.2 gives the overview of the information extracted from metadata of spatial data resources. The components of SDRP in the data model is detailed as follows:

- **Dataset spatial extent:** the geographic area covered by the dataset resources extracted from metadata description are stored in the data model $EX\_boundingPolygon$ class to allow extent based matching by the system.

- **Overview quality elements:** data quality overview elements are important for assessing the quality of a dataset for a particular application [3]. According to ISO quality evaluation principle, it is part of the indirect evaluation method [27]. Therefore, we need to profile information about the purpose, usage and description of the dataset in the data model $OvQualityElements$ class.

- **Data quality subelements**: used to assess the quality of spatial dataset for fitness for use based on users quality requirements. Therefore, quantitative quality description of spatial datasets needs to be extracted and populated in the data model $DQ\_Subelements$ class.

- **Spatial data resources:** this refers to the link or address of actual location of spatial dataset. If the dataset is spatial dataset, the resource locator defines the links, commonly expressed as Uniform Resource Locator (URL) [10]. The URL link enables users to obtain more information on the data resource. If the datasets are available online, unique identifier allows

to view or download the actual dataset. In addition, if the resource is a spatial data service, the locator defines the link, commonly expressed as a Uniform Resource Locator(s) (URL) to the service [49].

In the process of building SDRP the input is users spatial data search request including quality requirements. The output will be selected list of spatial data resources with detailed qualitative and quantitative quality values.The procedure of information extraction and selection of spatial data resource from metadata catalogue to our system SDRP is explained in the next section.

3.4.3 Spatial data resources extraction from metadata

In te process of populating SDRP spatial data resources can be filtered according to users application and spatial extent requirement into the system before further fitness for use evaluation. This phase of filtering spatial data resources is only the process of identifying the possible candidate datasets based on user application and extent requirement. Once required quality description of spatial data resource is populated into the SDRP, further evaluation will continue to determine fitness for use of the datasets.

In order to identify the metadata of spatial dataset, the first criteria we used is to search by users application requirement. The user application information can be found in different parts of he metadata such as: usage, purpose and description. Therefore, following the ISO metadata topic category concept [26], we defined theme keywords for user application to search datasets that can be used in relation to the user application.

Furthermore, in our system data model design, we decide to store the spatial extent information about dataset irrespective of what is inside the dataset for extent matching with user extent requirement. In order to extract and populate extent information of the dataset from metadata, the dataset and the system Spatial Reference System Identifier (SRID) should be the same. Because the SRID of the bounding box provided with the dataset my not be always the same with our system SRID. Therefore, SRID transformation is required. Also in order to use spatial functions to check the spatial extent matching, the user extent and the spatial dataset extent should have the same geometry type representation. We design the spatial data recommendation data model to store the extent information in the geometry type polygon. Therefore, the bounding box of the dataset extracted from metadata should be converted to polygon geometry in order to be stored in the data model.

To accomplish the spatial data resource extraction from metadata into the spatial data resources profile, we design algorithm 3. In the algorithm design we have used PostGIS spatial functions described below:

- $ST\_Polygon$: generates an ST_Polygon from a well-known text (WKT) representation and SRID. We use this function to convert the extracted extent information of the dataset into polygon
- $ST\_Intersects$: generates a boolean result after checking intersection between two geometry
- $ST\_Intersection$: takes two ST_Geometry objects and returns the intersection set as an ST_Geometry object.
- $ST\_GeomFromText$: returns a specified ST_Geometry to be enable the spatial function work
- ST_Area : returns the area of a polygon with double precision type

- $ST\_Transform$: transforms the ST_Geometry into the spatial reference specified by the spatial reference ID (SRID).

Variable definition used in Algorithm 3:

- U_E - user extent
- U_A - user application
- D_U - spatial dataset usage information (from overview quality element and topic category description)
- D_M - spatial dataset metadata
- D_E - spatial dataset extent from metadata (Bbox, SRID)
- BS_E - Extracted Bbox and SRID
- BS_E^t - dataset extent transformed
- DS_E - dataset extent the_geom
- $SDRP$ - spatial data resource profile
- I - 1 if U_E and DS_E intersect, 0 otherwise
- A_I - U_E and DS_E intersection area
- A_U - user extent area

Algorithm 3: Spatial data resource extraction from metadata catalogue

Procedure:

- search metadata from catalogue
- if user application is similar to usage information, extract Bbox and SRID from metadata and transform SRID to system SRID
- get geometry in the form of polygon (the_geom) from data extent transformed
- if there is intersection between user extent and dataset extent, compute the intersection area and the area of the user extent
- if the user extent and the dataset matching is above 25%, extract required information from metadata
- populate the spatial data resource profile

Input: $U_E, U_A, count = 0$

1: **while** U_A **do**
2: $D_M \leftarrow$ search_metadata()
3: $D_U \leftarrow$ extract_usage(D_M)
4: **if** $U_A \sim D_U$ **then**
5: $BS_E \leftarrow$ extract_D_E(D_M) /*store extracted D_E in temporary table*/
6: $BS_E^t \leftarrow$ ST_transform(ST_GeomFromText(BS_E),getsrid(system.the_geom))
7: $DS_E \leftarrow$ ST_PolyFromText(BS_E^t)
8: $I \leftarrow$ ST_Intersects(U_E, DS_E)
9: **if** $I = 1$ **then**
10: $A_I \leftarrow$ ST_area(ST_intersection(U_E, DS_E))
11: $A_U \leftarrow$ ST_area(U_E)
12: **end if**
13: **if** $\frac{A_I}{A_U} \geq 0.25$ **then**
14: $SDRP \leftarrow$ extract_info(M_D)
15: **end if**
16: **end if**
17: **return** $SDRP$
18: $count++$
19: **if** $count > 100$ **then**
20: break
21: **end if**
22: **end while**

3.4.4 Interaction profiling (IP)

Interaction profiling design are fundamental in recommendation service to reuse the users history. To design an effective interaction, one must consider what specific information are required to enhance the system operation [47]. Interaction profiling helps to understand users dataset usage history and contribute for the system to make better recommendation.

We define users interactions in the spatial data recommendation process, when users receive ranked list of recommended datasets based on their requirements and users access dataset(s) from the ranked list of spatial data resources. The interaction profile initialization and object creation is based on the users' spatial data retrieval and explicit feedback. This allows the system to perceive usability of a dataset and monitor datasets popularity as shown in figure 3.2. The datasets popularity information from IP can be used to inform users about the datasets how many times the datasets has been visited by other users. This will help users to be more informed about the datasets. At this point making use of others users dataset access history involves the collaborative recommendation techniques. Moreover, when spatial data users make use of the recommended datasets for their intended use and provide explicit feedback, IP build the actual usage of the datasets.

3.5 FITNESS FOR USE EVALUATION FUNCTIONALITY

After users spatial data search requirements and spatial data resources information are profiled in the spatial data recommendation data model, in order to recommend the spatial data resources for users, the system should make fitness for use evaluation. In this research we discuss the fitness for use evaluation from three aspect: spatial extent matching , application matching with spatial data resources description and overview quality elements, and quantitative data quality evaluation aspect. The order of fitness for use evaluation we choose to apply is extent matching, application matching and quality evaluation. However, since the fitness for use evaluation is performed using the system data model, the sequence does not have difference in recommending the datasets for users.

3.5.1 Fitness for use evaluation using spatial extent

Fitness for use evaluation using user spatial extent requirement requires extensive spatial matching to get dataset with the best fit extent. First of all the spatial data resources that have spatial extent matching with users spatial extent requirement needs to be filtered. For this purpose we design algorithm 4 to filter spatial datasets based on user spatial extent requirement. All the datasets which have intersection with user spatial extent requirements will be returned as a candidate dataset for further filtering.

In order to filter the candidate datasets by extent we compute the area ratio as given in algorithm 5. This phase of filtering spatial datasets needs to be addressed from different aspect of spatial extent matching functions. For example, the user extent requirement may be completely inside the dataset extent or only a portion of area of user extent may intersect with the dataset extent. Therefore, spatial area difference can be known by calculating the area ratio. Hence, area ratio computation of intersection with user spatial extent requirement and area ratio computation of intersection with spatial data resources extent helps to identify the best fit spatial data resources. The value of area ratio is given in percentage.

Then by sorting datasets descending using the ratio of intersection and user extent the system can identify and return the best datasets. If there are more datasets that have similar area ratio values, again the ratio of intersection and dataset extent help us to identify the best one. Based on

this logic we design algorithm 6 to rank spatial datasets using the computed area ratio.

In the algorithm design for the spatial computation the fallowing PostGIS built in functions are used:

- $ST\_GeometryTypee$ Return the geometry type of the ST_Geometry value.
- ST_within: returns true if one geometry is within the geometry of the other, it takes two arguments. We used the user extent and spatial dataset extent to return true or false.
- $ST\_Centroid$: This function takes one argument. We used it to return the centroid of the geometry given by the user as a point. Therefore, the centre of the user extent requirement can be check within the extent of dataset.
- $ST\_Intersects$: generates a boolean result after checking intersection between two geometry
- $ST\_Intersection$: takes two ST_Geometry objects and returns the intersection set as an ST_Geometry object.
- $ST\_GeomFromText$: returns a specified ST_Geometry to be enable the spatial function work

Variable definition used in Algorithm 4 - 6:

- U_A - user application
- $DS_S^i|_{i=1...N} \in DS_S$ - where DS_S is set of selected datasets
- $DS_E^j|_{j=1...N} \in DS$ - where DS is set of datasets
- I_E^i - user and dataset intersection extent
- A_I - U_E and DS_E intersection area
- R_{I_U} - A_I and area of U_E ratio
- R_{I_DS} - A_I and area of DS_E ratio
- DS_{SE}^i - selected extent dataset
- DS_{SS} - selected and sorted dataset
- DS_{SS}^S - sorted DS_{SS}

Algorithm 4: Select dataset based on user extent

Procedure:

- for all datasets, select a dataset if:
  - user extent is within dataset extent
  - center of user extent is within the dataset extent
  - user extent and dataset extent intersection has polygon geometry
  - return DS_S

Input: DS, U_E, DS_E

for DS^i **to** DS^N **do**
  $DS_E^i \leftarrow$ extract_dataset_extent(DS^i)
  if ST_within(U_E, DS_E^i) **then**
    $DS_S \leftarrow DS^i$
  else if ST_within(ST_centroid(U_E, DS_E^i)) **then**
    $DS_S \leftarrow DS^i$
  else
    if ST_Intersect(U_E, DS_E^i) **then**
      $I_E^i \leftarrow$ ST_Intersection(U_E, DS_E^i)
      if ST_GeometrType(ST_GeomFromText(A_I)) = "ST_Polygon" **then**
        $DS_S \leftarrow DS^i$
      end if
    end if
  end if
end for
return DS_S

Spatial datasets which have a polygon intersection with user spatial extent requirement are returned as a result of algorithm 4. Once the spatial datasets are filtered by the spatial extent matching as given by algorithm 4, the selected datasets will be an input for algorithm 5.

Algorithm 5: Area ratio computation

Procedure:
- for all selected datasets:
  - calculate user extent and dataset extent intersection
  - compute intersection and user extent area ratio
  - compute intersection and dataset extent area ratio
  - return dataset selected, area ratios

Input: DS_S, U_E, DS_{SE}

for DS_S^i **to** DS_S^N **do**

$I_E^i \leftarrow$ ST_Intersection(U_E, DS_{SE}^i)

$R_{I\_U}^i \leftarrow \frac{ST\_Area(I_E^i)}{ST\_Area(U_E)}$

$R_{I\_DS}^i \leftarrow \frac{ST\_Area(I_E^i)}{ST\_Area(DS_S E^i)}$

end for

return DS_S, R_{I_U}, R_{I_DS}

Algorithm 5 returns the same dataset that has been returned by algorithm 4 with newly computed area ratio extent information. This area ratio helps to order the dataset in order to identify the best one. The ranking procedure using spatial extent ratio value for every candidate spatial dataset is given in algorithm 6. However, for simplicity purpose we use sort function supported in PostGIS for implementation as given in algorithm 7.

Algorithm 6: Rank dataset based on Extent ratio

Procedure:

- sort selected dataset by R_{I_U} (A_I and area of U_E ratio)
- for all selected and sorted datasets, if two or more consecutive datasets have equal R_{I_U}, sort these rows with R_{I_D} (A_I and area of DS_E ratio) else update the index to indicate to the next group

Input: DS_S, R_{I_U}, R_{I_D}

1: $DS_{SS} \leftarrow$ sort DS_S desc R_{I_U}
2: **for** i **to** M **do**
3: //where M is the number of selected and sorted datasets
4: **if** $R^i_{I\_U} = R^{i+1}_{I\_U}$ **then**
5: **for** $j = i$ **to** M **do**
6: **if** $R^i_{I\_D} = R^j_{I\_D}$ **then**
7: $temp \leftarrow DS^j_{SS}$ {temporarily save current record DS^j_{SS}}
8: //next two lines swap current record with the next
9: $DS^j_{SS} \leftarrow DS^{j+1}_{SS}$
10: $DS^{j+1}_{SS} \leftarrow temp$
11: **end if**
12: //to check the next group having the same R_{I_U}
13: **if** $R^{j+1}_{I\_U} \neq R^{j+2}_{I\_U}$ **then**
14: //if the next two records A_I and area of U_E ratio are not equal,
15: //set position for the next comparison to this group and break the
16: //inner loop
17: $i = j + 2$
18: break
19: **end if**
20: **end for**
21: **end if**
22: **end for**
23: **return** DS^S_{SS}

Algorithm 7: Rank dataset based on Extent ratio (implementation version of algorithm 6)

Procedure:

- sort selected dataset by R_{I_U} (A_I and area of U_E ratio)
- for all selected and sorted datasets, if two or more consecutive datasets have equal R_{I_U}, sort these rows with R_{I_D} (A_I and area of DS_E ratio)
- return DS^S_{SS}

Input: DS_S, R_{I_U}, R_{I_D}

1: $DS^S_{SS} \leftarrow$ sort DS_S desc R_{I_U}, desc R_{I_D}
2: **return** DS^S_{SS}

3.5.2 Fitness for use evaluation using application

To design the fitness for use evaluation based on user application requirement, it is required to define theme_keyword that represent the application by referring thematic classification of dataset. The concept of theme_keyword definition is mainly required to search different spatial datasets which can be useful for the intended application. The theme_keyword definition for the application gives wide range of possibly to search various resources for the intended application. The ISO standard metadata representation topic categories is one of metadata elements required to identify a dataset, that is used to group keywords and to learn more about main themes of the dataset to understand topics exist in the dataset description. It is high-level geographic data thematic classification to assist in the grouping and search of available geographic datasets [10]. The topic category is also used for topic-based search of available spatial data resources. It is one of a handful element that describes the type of features that are included in a dataset.

Datasets that come from different sources with different theme description can be useful for an application. The theme describes the type of features that are included in a dataset. Also, the theme description indicates for what purpose the dataset can be used for. Therefore, to facilitate the spatial data search based on user application requirement, we also need to define set of theme keywords. This will enable the recommendation system to search more datasets that can be used for user application.

Here we address fitness for use reasoning logic to recommend spatial dataset based on user application and predefined theme_keyword. Our consideration for the application based fitness for use evaluation logic design is that all datasets which have exact matching to the user application have higher weight than datasets which are returned based on theme_keyword.

First we design algorithm 8 to search datasets based on similarity of user application requirement and corresponding theme_keywords. This algorithm identifies the application name specified by the user and the corresponding theme_keyword from the system. Then, it makes keyword matching of application name and the theme_keyword with overview quality element and description of spatial dataset. Then all set of spatial datasets that can be recommended for the user applications are returned. Once these datasets are ready algorithm 9 continue to quantify values for exact application name and the theme_keywords matching found in the datasets. In this algorithm design the assumption is the exact matching of user application name in the dataset has higher weight than the summation of the corresponding theme_keywords. This will help us to rank the dataset which are more relevant to user application. After the quantity of keywords found in the candidate datasets are identified, algorithm 10 computes the total sum of the theme_keywords found in each datasets. Finally algorithm 12 is used to rank the datasets using the weight assigned for the application and the theme_keyword.

Variable definition used in Algorithm 8-12:

- U_A - user application
- $DS_S^i|_{i=1...N} \in DS_S$ - where DS_S is set of selected datasets based on U_A or TKW
- $TKW_j|_{j=1...M} \in TKW$ - where TKW is set of theme_keywords
- $\Omega_{TKW}^j \in \{0, 1\}$ - weight assigned for theme_keyword j
- Ω_A - weight assigned for U_A
- DS - dataset

- O_Q - overview quality description of DS^i
- $N = |D_S|$ - number of datasets D_S in database
- M - number of theme_keyword of an application
- $S_i \in S$ - where S is the sum of theme_keywords in DS_S
- w total number theme_keywords defined for application
- R_i percentage value of relevance based on application and TKW similarity found

Algorithm 8: Select datasets using user application and theme_keywords

Procedure:
- for all datasets select a dataset if:
  - user application and the theme_keyword is similar to overview quality description of the dataset or
  - the theme_keyword is similar to overview quality description of the dataset even if user application is not similar to overview quality description of the dataset.

Input: U_A, TKW, DS

1: **for** $i = 1$ **to** N **do**
2: **if** $U_A \sim O_Q$ **then**
3: **if** $TKW \sim O_Q$ **then**
4: $DS_S \leftarrow DS^i$
5: **end if**
6: **else**
7: **if** $TKW \sim O_Q$ **then**
8: $DS_S \leftarrow DS^i$
9: **end if**
10: **end if**
11: $i \leftarrow i + 1$
12: **end for**
13: **return** DS_S

The result of algorithm 8 is the set of datasets that the overview quality or description has matching with user application or the corresponding theme_keywords. This datasets used in the process to quantify the application and the theme_keywords matching found in the dataset as shown in algorithm 9:

Algorithm 9: Quantify application name and theme_keyword in DS_S

Procedure:

- for all datasets, compare user application with each overview quality description of the dataset. When ever they are similar, set weight of the application as the sum of all the theme_keywords, otherwise set the weight to 0
- for all datasets and for all theme_keyword, if a theme_keyword is similar to overview quality description of the dataset, set the theme_keyword weight to 1, else 0
- return weight assigned for user application and weight assigned for theme keyword

Input: DS_S, U_A, TKW

1: **for** DS_S^i **to** DS_S^N **do**
2: $O_Q^i \leftarrow$ get_overview_quality(DS_S^i)
3: **if** $U_A = O_Q^i$ **then**
4: $\Omega_A \leftarrow \omega \mid \omega > \sum_{j=1}^{M} \Omega_{TKW}^j$
5: **else**
6: $\Omega_A \leftarrow 0$
7: **end if**
8: **for** TKW_j **to** TKW_M **do**
9: **if** $TKW_j = O_Q^i$ **then**
10: $\Omega_{TKW}^j \leftarrow 1$
11: **else**
12: $\Omega_{TKW}^j \leftarrow 0$
13: **end if**
14: **end for**
15: **end for**
16: **return** Ω_A, Ω_{TKW}

The output of algorithm 9 is conversion of subjective matching into quantitative values. As explained before, the weight given for the user application matching should be greater than sum of all the theme_keywords defined for that specific application. This enables to identify datasets that match user application in prior than other datasets selected as candidate datasets based on theme_keywords. The theme_keywords are assigned boolean values as weight to indicate matching is found or not in the dataset where a weight equal to 1 means that matching has been found. This values again summed up using algorithm 10 as shown below:

Algorithm 10: Compute sum of theme_keywords in dataset

Procedure:

- compute sum of theme_keywords in dataset DS_S^i theme_keyword

Input: DS_S, Ω_{TKW}

1: **for** DS_S^i **to** DS_S^N **do**
2: $S_i \leftarrow \sum_{j=1}^{M} \Omega_{TKW}^j$
3: **end for**
4: **return** S

When the process of selecting spatial dataset based on user application, searching datasets by theme_keywords and assigning weight value completed, these values are used to rank spatial datasets that best fit user application. We said that application similarity found in the spatial dataset overview quality element has higher priority than other datasets. Also a datasets that

have more theme_keyword matching has second priority. Based on this assumption we devise a relevance indicator to inform users how much percent a dataset fits their application. In order to elaborate our approach we assume that there is an application with five theme_keywords and designed algorithm 11 as shown below:

Algorithm 11: Display relevance of DS_S based on application and theme_keywords weight

Procedure:

- For each datasets DS_S filtered by U_A and TKW
- Extract the weight Ω_A and S
- If exact matching for user application and the dataset overview quality, then indicate dataset is 100% relevant
- Otherwise calculate the difference between w and s^i to indicate the corresponding relevance

Input: DS_S, w

for DS_S^i **to** DS_S^N **do**
  $\Omega_A \leftarrow$ get_weight_by_$UA(DS_S)$
  $S \leftarrow$ get_weight_by_$TKW(DS_S)$
  if $\Omega_A^i = w$ **then**
    $relevance \leftarrow Ri\%$
  else if $w - s^i = 1$ **then**
    $relevance \leftarrow Ri\%$
  else if $w - s^i = 2$ **then**
    $relevance \leftarrow Ri\%$
  else if $w - s^i = 3$ **then**
    $relevance \leftarrow Ri\%$
  else if $w - s^i = 4$ **then**
    $relevance \leftarrow Ri\%$
  else
    $relevance \leftarrow Ri\%$
  end if
end for

Once the final result for application matching is returned from algorithm 11 the relevance indicator can be used to order the datasets as shown in algorithm 12.

Algorithm 12: Rank selected datasets DS_S based on application and theme_keywords

Procedure:

- sort the dataset according to application and sum of theme_keywords

Input: DS_S

$\Omega_A \leftarrow$ get_app_weight(DS_S)
$S \leftarrow$ get_sum_theme_keyword(DS_S)
$DS_{SS} \leftarrow$ sort DS_{SS} desc Ω_A, desc S
return DS_{SS}

3.5.3 Fitness for use evaluation using quality element

To design the fitness for use evaluation of spatial datasets based on quality elements, the spatial datasets need to be extracted based on user extent and application requirement and populated in the system profile as we discussed in section 3.4.3. Once the datasets quality description and necessary information are populated in the system, fitness for use evaluation can be performed. In order to compute the fitness of a dataset by comparing the quantitative data quality elements, the measurement unit of each quality element should be adjusted into the same measurement unit.

In this section we address the process of computing the fitness for use evaluation using quantitative data quality elements. to recommend a spatial datasets based on users quality requirements, we design algorithm 13 to compute range for all user quality requirement values. This is because it is not always possible to find spatial dataset that exactly match users requirement. We decided to subtract and add half of user quality requirement value to a user quality requirement value itself for each element to set the minimum and maximum of range.

This approach does not work always for all spatial data quality elements. Therefore, it is required to consider special cases to determine ranges. For example if the user quality requirement is zero percent error it is not possible to determine the minimum and the maximum range from the user quality requirement by subtracting and adding half value. Also for positional accuracy requirement once user provide quality requirement both +/- of root mean square error (RMSE) should be considered. Therefore, the logic to compute range for user quality requirement should cover all separately. Our approach for cases with the zero error user requirement, we set constant minimum and maximum value. And for positional accuracy we modify the approach to consider plus and minus of RMSE in the process of rang computation as shown in algorithm 13.

After the minimum and the maximum value of the user quality requirements is computed as a range to maximize the number of candidate datasets, we design algorithm 14 to fetch all the datasets quality subelements based on the computed range and return boolean result to indicate the spatial data quality of each element is within the range or not: a boolean value one means the dataset quality value is within the range; otherwise boolean value will be set to zero.

Then we design algorithm 15 to multiply the boolean values of the dataset quality elements with the user quality requirements weights. And algorithm 16 sum up weighted dataset quality elements which shows the relevance of a dataset based on user quality requirement weights.

From the result of algorithm 16 more datasets can have equal relevance based on users quality requirement. Therefore, to determine the best one from these datasets, we design algorithm 17 to compute the distance of each data quality element of the dataset with respect to the user quality requirement value. Finally we design algorithm 18 and 19 to rank the datasets. Variable definition used in Algorithm 13:

- Q_R - set of user quality subelement requirement
- Q_N - user quality subelement name
- E_P - Absolute positional accuracy
- G_P - Grided positional accuracy
- I_P - Relative positional accuracy
- $Q^i_{N\_min}$ - minimum range of ith quality subelement
- $Q^i_{N\_max}$ - maximum range of ith quality subelement

- Q_{R+} - positive Q_R^i
- Q_{R-} - negative Q_R^i
- r - range calculated based on user Q_R

Algorithm 13: Calculate range based on user quality requirements

Procedure:

- for all user quality subelement requirement:
  - if each user quality subelement name is not similar to positional accuracy:
    - for all user subelement quality whose measurement unit is percentage, if user quality subelement has value zero, set its minimum to 0 and it maximum to 2. If it has value 1, set the minimum to 0 and its maximum to $\frac{3}{2}$ of user quality subelement requirement. Otherwise set minimum to $\frac{1}{2}$ of user quality subelement and maximum to $\frac{3}{2}$ of user quality subelement requirement.
  - if each user quality subelement name is similar to positional accuracy:
    - if user quality subelement value is zero set the minimum and maximum to -0.5 and $+0.5$ respectively. Else set the minimum to $-\frac{1}{2}$ of the user requirement and maximum to $\pm\frac{3}{2}$ of user quality subelement
- return r

Input: Q_R, Q_N, E_P, G_P, I_P

for Q_R^i **to** Q_R^N **do**
  if $Q_N^i \nsim E_P || G_P || I_P$ **then**
    if $Q_R^i = 0$ **then**
      $Q_{N\_min}^i \leftarrow Q_R^i$
      $Q_{N\_max}^i \leftarrow Q_R^i + 2$
    else if $Q_R^i = 1$ **then**
      $Q_{N\_min}^i \leftarrow Q_R^i - Q_R^i$
      $Q_{N\_max}^i \leftarrow \lceil Q_R^i + \frac{1}{2} Q_R^i \rceil$
    else
      $Q_{N\_min}^i \leftarrow \lceil Q_R^i - \frac{1}{2} Q_R^i \rceil$
      $Q_{N\_max}^i \leftarrow \lceil Q_R^i + \frac{1}{2} Q_R^i \rceil$
    end if
  else
    if $Q_R^i = 0$ **then**
      $Q_{N\_min}^i \leftarrow Q_R^i - 0.5$
      $Q_{N\_max}^i \leftarrow Q_R^i + 0.5$
    else
      $Q_{R+} \leftarrow Q_R^i$
      $Q_{R-} \leftarrow -Q_R^i$
      $Q_{N\_min}^i \leftarrow \frac{1}{2} Q_{R-} + Q_{R-}$
      $Q_{N\_max}^i \leftarrow \frac{1}{2} Q_{R+} + Q_{R+}$
    end if
  end if
  $r \leftarrow [Q_{N\_min}, Q_{N\_max}]$
end for
return r

The range computed from algorithm 13 are used as a criteria to identify data quality elements of a dataset that satisfy user quality requirements in algorithm 14. Specifically algorithm 14 assigns boolean values to spatial data quality if the datasets are with in a range

Variable definition used in Algorithm 14 - 18:

- Q_R - set of quality range
- $DS_i \in DS$ - where DS is set of datasets
- $Q_{DS}^{ji} \in Q_{DS}^{j}$ - is the ith quality of the jth dataset where Q_{DS}^{j} is set of quality of dataset DS_i
- r - range calculated based on user Q_R
- Q_w - set of weight of quality elements provided by user
- $d_{ji} \in D$ - where D is set of distance between dataset quality and user quality
- X - 1 if DS is in r, 0 otherwise
- X^w - weighted X
- $S_j \in S$ - sum of X_j^w where S is set of sum of X^w

Algorithm 14: Evaluate data quality of dataset based on user quality range

Procedure:
- for all datasets, for all user quality subelements and all dataset quality subelement:
  - get the minimum and maximum ranges of the user quality subelements
  - if dataset quality subelement is with in these range, then set a boolean variable X to 1, otherwise 0
- return X

Input: DS, r, Q_R

for DS_j **to** DS_M **do**
  $//N$ is the number of quality elements
  for Q_R^i **to** Q_R^N **do**
    $//M$ is the number of datasets
    for Q_{DS}^{ji} **to** Q_{DS}^{jN} **do**
      $[Q_{min}^i, Q_{max}^i] \leftarrow$ fetch(r)
      if $Q_{min}^i < Q_{DS}^{ji} < Q_{max}^i$ **then**
        $X_{ji} = 1$
      else
        $X_{ji} = 0$
      end if
    end for
  end for
end for
return X

The boolean result from algorithm 14 will be multiplied by the user quality requirement weight as shown in algorithm 15:

Algorithm 15: Weighted X (dataset data quality subelements relevance to user quality subelements)

Procedure:

- for all datasets:
  - fetch boolean value corresponding to each dataset quality subelements
  - for all user quality subelements:
    - fetch user quality subelements weight
    - multiply each user quality subelements weight by dataset quality subelements boolean value X
- return X

Input: DS, Q_R, Q_w

1: **for** DS_j **to** DS_M **do**
2: $X \leftarrow$ get_boolean(DS)
3: **for** Q_R^i **to** Q_R^N **do**
4: $Q_w^i \leftarrow$ get_user_quality_weight(Q_w)
5: $X_{ji}^w = X_{ji} Q_w^i$
6: **end for**
7: **end for**
8: **return** X^w

At this point the spatial datasets of each quality elements are evaluated according to user quality requirement and the result is multiplied by the user assigned weight. Now to determine the relevance of the dataset, the individual quality element weighted value should be summed up. Algorithm 16 aggregates all the quality elements to compute the dataset relevance indicator as shown bellow:

Algorithm 16: sum of weighted X (dataset relevance to user quality requirement)

Procedure:

- for all datasets weighted boolean value, compute sum of weighted X
- return sum weighted X

Input: $DS, X^w, Q_{DS}, S_j = 0$

1: **for** DS_j **to** DS_M **do**
2: $X^w \leftarrow$ get_weighted_boolean(DS)
3: **for** X_{ji}^w **to** X_{jN}^w **do**
4: $S_j \mathrel{+}= X_{ji}^w$
5: **end for**
6: **end for**
7: **return** S

From algorithm 16 it is possible to get spatial datasets which have similar relevance indicator. This happens because spatial dataset quality value is considered as fit if it fall with in the computed range from algorithm 13. Therefore, in order to identify the best fit datasets we choose to compute the distance of each dataset quality element from user quality requirement in algorithm 17.

Algorithm 17: Calculate distance of dataset quality from user quality

Procedure:

- for all datasets:
  - fetch boolean values of the datasets quality elements
  - for each user quality requirements
    - for a single dataset, if each boolean value is true, then compute the distance from user quality subelement to dataset quality subelement

Input: DS, Q_R, Q_w

for DS_j **to** DS_M **do**
  $X \leftarrow$ get_boolean(DS)
  for Q_R^i **to** Q_R^N **do**
    for X_{ji} **to** X_{jN} **do**
      /*jth dataset and i to N quality subelements*/
      $X_{ji} =$ fetch(X) /*X - boolean value*/
      if $X_{ji} = 1$ **then**
        $d_{ji} \leftarrow |Q_R^i - Q_{DS}^{ji}|$
      end if
    end for
  end for
end for
return D

Finally using the computed relevance indicator from algorithm 16 and the distance value of each quality element of the dataset from algorithm 17, it is be possible to identify the best dataset that fits user quality requirements. Therefore, we design algorithm 18 and 19 to rank the datasets that best fit to the user quality requirements.

Algorithm 18: Rank datasets DS by relevance based on quality element evaluation

Procedure:

- sort the dataset according to their aggregated weighted boolean X

Input: DS

$S \leftarrow$ get_sum_weight_boolean(DS)
$DS_S \leftarrow$ sort DS desc S
return DS_S

Variable definition used in Algorithm 19:

- DS_S - sorted datasets
- DS_{SS} - sorted DS_S by distance
- Q_w - set of weight of quality elements provided by user
- Q_w^{max} - the maximum weight of quality elements provided by user
- Q_w^{name} - the name of the maximum weighted of quality elements
- D - distance between dataset quality and user quality
- d_{ji} - is distance between dataset quality and user quality in ith column

Algorithm 19: Rank dataset by identifying relevance by distance

Procedure:
- Input datasets sorted by relevance value:
-for all user quality requirement weight
  - identify the maximum user quality requirement weight assigned
  - identify the name of quality element which have maximum weight
  - find the dataset quality element which have the same name
  - for each DS_S identify the distance value of quality element
    - sort the DS_S based on the distance value

Input: D, Q_w, DS_S

for Q_w^i **to** Q_w^N **do**
  $Q_w^{max} \leftarrow$ fetch_max(Q_w)
  $Q_w^{name} \leftarrow$ get_name(Q_w^{max})
  if $DS_S^{Q\_name} \sim Q_w^{name}$ **then**
    for DS_S^j **to** DS_S^M **do**
      $d_{ji} \leftarrow$ fetch(DS_S)
    end for
  end if
  $DS_{SS} \leftarrow$ sort(DS_S) desc d_{ji}
end for
return DS_{SS}

3.5.4 Profile update functionality

As we discussed in section 3.4 recommendation system profile update is the core functionality to search and recommend spatial dataset based on user recent spatial data search quality requirements. In our recommendation system design the profile update depends on users. System users may use the system to browse datasets for their intended application by changing the quality requirements. Therefore, the system profiling should have to keep the recent information about users specific quality requirements for specific application. Also when users accesses spatial data resources from the recommendation list, the system follow users click event on the dataset and build information about frequency of dataset access history and update profiles. This information will help to inform subsequent users how frequently the dataset has been visited by other

users. To get information about spatial data actual usage and the recommendation facility users are expected to provide feedback. This enables the system in updating the spatial dataset actual usability.

Algorithm shows user profile update when users provide their spatial data search quality requirements

Variable definition used in Algorithm 20:

- Q_R - user spatial data search quality requirements.
- U_C - Current user

Algorithm 20: Update user profile

Procedure:
- if user spatial data search quality requirement exists with current user
  - update quality requirement
  - otherwise insert user requirement into user profile

Input: Q_R, U_C

if U_C Exist **then**
  $UP \leftarrow$ update_UP(Q_R)
else
  $UP \leftarrow$ insert_into_UP(Q_R)
end if

Based on user profiled information the system generates recommendation dataset for users. When users access the dataset the system build interaction profile by tracking user dataset access history. Also the system allow users to provide their feedback about the actual dataset usage. Algorithm 21 shows the process of updating system profile.

Variable definition used in Algorithm 21:

- DS_R - recommended datasets
- DS_P - picked dataset
- U_R - user spatial data search requirement.
- $count$ - count for dataset used for same application
- F_B - spatial dataset usability feedback

Algorithm 21: Update spatial data resources and interaction profile

Procedure:
- if the user click on the recommended dataset
    - update dataset count
    - if the user provide feedback about dataset
        - store the feedback in the interaction profile

Input: DS_R, U_R

$UP \leftarrow$ update_UP(U_R)
$DS_P \leftarrow$ identify_dataset_picked(DS_R)
if DS_P **then**
  $DSCount \leftarrow$ update_DS_count
  $count \leftarrow count + 1$
end if
if F_B **then**
  $I_P \leftarrow$Update_usability_DS
end if

3.6 SUMMARY

In this chapter we explained the proposed spatial data recommendation system data model design and the reasoning logic. Starting with the system architecture that gives a general description of the system, we explained the possible techniques and methods that can be used to initialize and represent the profile. The required information to build spatial data recommendation profile: user spatial data search quality requirements, spatial data quality description and interaction are elaborated specifically.

We designed Algorithm 1 and 2 for checking users spatial data search quality requirements, and Algorithm 3 to retrieve spatial data resources. These algorithms shows the activities of the recommendation system profile initialization and representation. And its output is an input for the fitness for use evaluation process in the system. Therefore the recommender system will make fitness for use decision based on this profiled information. Algorithm 4 - 21 details the process of fitness for use evaluation procedures.

Next we design the fitness for use reasoning logic to determine the best fit datasets based on user spatial data search requirement. In the fitness for use reasoning logic we address spatial matching, user application matching and user quality subelemenets evaluation. In each section of the fitness for use evaluation design we explained about how the reasoning logic design is designed. We also provide the procedures for each algorithm. In the next chapter we addressed the recommendation system functional units design and explanation about the selected case study for prototype implementation.

Chapter 4
Spatial data recommendation system design

4.1 INTRODUCTION

This chapter focuses on the recommendation system functionality and the explanation on a selected use case. The proposed spatial data recommendation system is aimed at delivering spatial data to user based on fitness for use. To accomplish this, several functionalities need to be designed. We show the main functional requirements of the spatial data recommendation system and how all the functional units integrated to accomplish best fit dataset delivery objective. In this chapter we introduce the selected case study to reflect importance of the fitness for use evaluation in the selected application. The prototype implementation of fitness for use reasoning logic designed in chapter 3 section 3.5 is based on the case study explained in this chapter.

4.2 SYSTEM FUNCTIONAL REQUIREMENTS

In order to recommend spatial dataset based on fitness for use evaluation using user profile information, the recommendation system needs to support various functionalities. Those functionalities enable users to search spatial datasets according to their requirements and get information about how it fits their need.

To search spatial data based on fitness for use, first the system needs to identify the user. This helps the system to learn user quality requirements and to recommend data resources for the respective user. As we explained in section 3.4 the spatial data recommendation system is working based on profiling. At a time when a user logged into the system, it identifies the user and opens a session. Once the system identifies current user through opened session, the system offers functionalities that allow users to provide their spatial data search requirements to search spatial data resources. The system stores the users quality requirements into users profile (UP). This will help to identify the users interest in recommendation process.

The recommendation system allow users to search spatial data by providing complete spatial data search requirements. The complete search requirements include spatial extent, application and quality requirements with weight. Also the system allow users to search dataset by specifying partial search requirement. It means that the system allow users to search dataset by extent together with application requirement or by extent together with quality requirements and weight. Therefore, based on the user input the system evaluate fitness for use and recommend datasets that best fits users requirements.

If users provide complete spatial data search requirements including extent, application and quality requirement, the system search and determines the fitness for use based on the user search requirements in a sequence of extent matching, application matching and quality value comparison as explained in 3.5. In case where users provide partial data search requirements, the system makes fitness for use evaluation and dataset recommendation according to the user input.

Users may require to know the level of fitness for use of the dataset for their requirements. Since they provide extent requirement, application and quality requirements, the fitness for use

of the dataset for user requirements is different per each of fitness for use evaluation criteria. For example, the dataset which is best fit with the user extent requirement may be less fit by the quality element evaluation. The fitness for use evaluation based on the user quality requirements has different value per each quality element value as explained in algorithm 17. Thus, the user may require to know which dataset quality satisfies which specific quality requirement. Therefore, the system should support a mechanism to allow user to select ranking criteria to identify which dataset best fit particular requirement. The ranking allow user to get the best fit dataset on top of the recommendation list. The ranking functionality of the system includes rank by extent, rank by application, and rank by quality requirement.

In addition to the ranking functionality, the system inform users how much percent of the requirement is satisfied. This helps in cases where there are more recommendation datasets for users with similar fitness for use result. Users will not depend on the order of the recommendation list to differentiate the best one if they are informed by the value of level of fitness for use. Moreover, by changing the ranking criteria users can easily identify the dataset that have maximum fit to their requirement.

4.2.1 Components of the recommendation system

The recommendation system is the core part of the overall system realization of the recommendation system architecture introduced in section 3.2. The recommendation system includes different functional units to accomplish automated and complete search for spatial data based on fitness for use. The use case diagram in figure 4.1 shows the overall actions of the recommendation system. The diagram focuses on the system processes, user and system functionalities thereby pictorially representing what the system can do to determine spatial datasets fitness for use. Moreover, the use case shows the overview on how one functional unit makes use of other functionalities to achieve the final goal.

Different functional units are used for construction of spatial data recommendation profile, evaluation of fitness for use of a dataset using user requirement from profile, ranking the datasets and recommending them based on user requirement. These system functionalities are:

- **User registration and login**: system allows registration for first time users. The user registration is used in the system to create initial profile for a user. The system use the unique information in registration to identify users during recommendation process and to update system profile. If users are already registered, they can use the system login functionality. Once a user is logged into the system a session will be opened based on user unique identifier which enables the system to identify current user.

- **Specify user requirements**: in order to evaluate datasets fitness for use, users should specify their spatial extent, their application information and the quality requirements with corresponding weights. Therefore, the recommendation system design should have to support user interface for users to provide their quality requirements. Once users submit their requirements the system checks the validity of user input before storing it in the system. This functional unit supports users input validation as explained in section 3.4.1.

- **Retrieval of spatial data resources**: this functionality is used to find spatial data resources that fulfil users spatial data search requirements. To search spatial data and recommend it to users based on fitness for use, retrieval of the data resources is important functionality. This service is used to search spatial data directly from catalogue based on users specific search requirements as explained in section 3.4.3. In addition, the spatial data retrieval

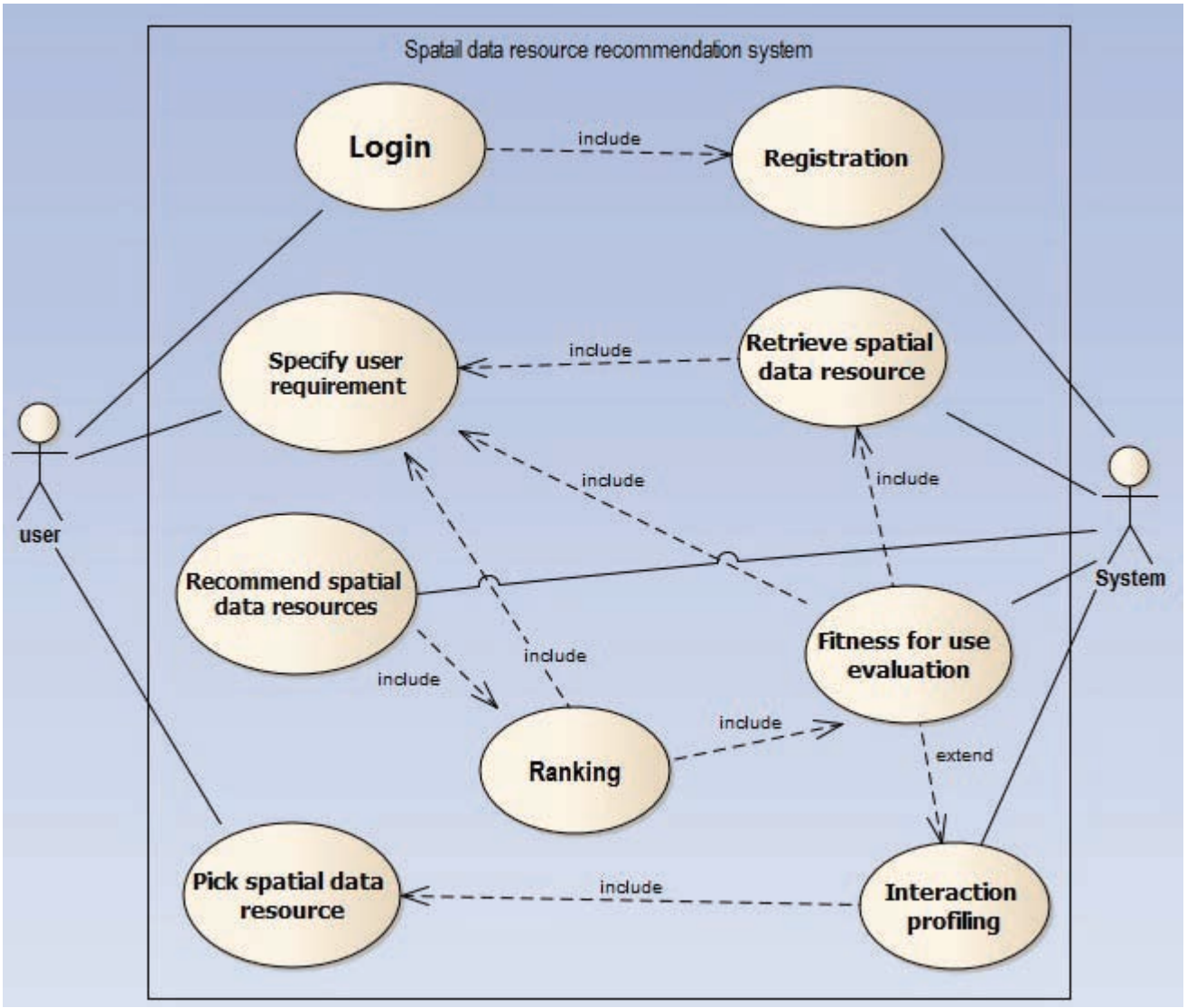

Figure 4.1: Recommendation system use case diagram

functionality enables the system to retrieve the datasets which are stored in the database for fitness for use evaluation.

- **Fitness for use evaluation**: this function is the core part for spatial data recommendation system that is based on fitness for use. It performs the fitness for use evaluation of spatial datasets according to users spatial data search quality requirements. The fitness for use evaluation operation includes extent matching, application matching, and comparison of users quality requirements and quantitative quality subelements of spatial datasets as explained in section 3.5. Once fitness for use evaluation of the spatial datasets based on user requirements is completed, the unsorted list of recommendation datasets pass to the ranking service.

- **Ranking service**: this service identifies ranking criteria selected by the user and rank the dataset accordingly. In the spatial recommendation system users can choose to rank either by extent, application name or by quality as explained in Algorithm 12, Algorithm 7, Algo-

rithm 18, and Algorithm 19. When user choose the ranking method, the system first orders the datasets according to the selected ranking criteria. Therefore user can easily get the best fit data for their requirement on top.

- **Recommendation of spatial data resources**: this service delivers datasets that meets requirements based on fitness for use evaluation and display the recommendation datasets according to user selected ranking method. It is a system service to let users get access to the recommended spatial dataset.
- **Picking a dataset**: it is a functional unit to identify the spatial dataset which users selected. This functional service used to update the recommendation system profiles. After the system returned recommendation datasets, and users picked a dataset, the system make use of the users selection event to build usability information using this functional unit. However, the actual use of the spatial datasets should be learned from user explicitly.
- **Interaction profiling**: this function works with the integration of other functional units of the system. Once the system learns users interaction to the system, it store the user requirement used to search the dataset in user profile and it creates or update usability information about dataset in system interaction profile.

4.3 SYSTEM NON FUNCTIONAL REQUIREMENTS

In this thesis work we focused on testing the spatial data fitness for use reasoning logic with simple web application system.

- We provide spatial data recommendation system as web based application with simple interface to facilitate easy system access for all users with user login service.
- Secured system based on login name and password credential for opening a session and identifying user's spatial data search requirement.
- Display the dataset extent on open layer map based on user selection to enable users to see the extent variation between the recommended dataset and their extent requirement.
- Display the value used to rank the recommendation to allow users to choose the dataset based on relevance value than based on order of display.

4.4 USE CASE DEFINITION

In order to implement the fitness for use reasoning logic as a prototype for demonstration purpose we selected a use case. The case study we selected focuses on searching spatial dataset for taxation and planning application. We select these two application, because they require spatial datasets. Hence, we can use them to elaborate the reasoning logic we designed in this thesis work.

In planning and taxation application spatial data resources are used for resource management and developmental planning by different organizations like private companies, federal, states and local management agencies. City and county officials use spatial data resources to establish tax, zoning, and emergency response districts. The real estate industry uses spatial data resources to determine the best location for a new business plan based on area population and land values. City planners use spatial data resources for replacement of sidewalks, the initial paving or repaving of streets, the replacement of water mains etc. Foresters use spatial datasets to identify potential forest land purchases to plan future forest expansion.

All these users require spatial data with different level of detail and quality requirement. In spatial planning information about existing objects in the real world are required. Also the implementation of spatial planning requires integration of different spatial datasets and information about land owners. It means in addition to spatial data used for planning, information on land ownership and the spatial extent of the land owned by a specific owner is required [35]. Geographical information systems like cadastre and land register are important in planning to provide this form of information.

Spatial data is used for a wide range of transportation planning including accident analysis, transportation demand modelling, infrastructure management, transportation policy analysis, commercial vehicle operations, transit operations, and intelligent transportation systems [18]. In these applications the value of any spatial data depends more on its fitness for a particular use. Transportation planning requires additional data which is related to geography like traffic density, street capacity, public transport routes etc. Using these data, planners may want to know a specific set of road network characteristics that are present at a given location.

Moreover, transportation planning agencies use spatial data to locate or describe events on a transportation system which involve the requirement of up to date resources. Geographically referenced transportation question requires spatial analysis that combines information including road geometry and intersection topology, information that describes the characteristics of the transportation system and objects that are found around the road [5]. Road expansion related questions requires spatial analysis about the closeness of the utility to buildings, correctly identifying the location, intersection and under/over pass network lines etc. Therefore, evaluating the completeness of spatial data, area of interests, accuracy of object measurement, time related information of the dataset is important.

Spatial planning of land price is also one of the application that requires spatial data integration based on user quality requirement for intended use. Taxation authority need spatial data for taxation policy making and to analyse factors that influence tax computation and their price [41]. For such usage spatial data resources need to be evaluated and their quality need to be better understood based on user specific quality requirement.

Factors that drive urban and residential tax computation are highly correlated to spatial data resources. Geographic location has an important overall roll for taxation [41]. Underlying driving factors and tax calculation methods are not the same from region to region and from one type of land to other type of land. However, it is directly linked to spatial location and information about property size, housing, landscape, environmental situation, and land use informations etc. Moreover, full and timely description of rights of subjects on objects is also important for taxation, financial institution process and planning. Therefore, accuracy in geometrical measurements, attribute information observations, correct area estimation, checking the dataset logical consistency and time measurement are important for using the dataset in fair taxation and estimation of financial planning.

Taxation authorities determine tax using the context of land use and area size [41]. Every time when there is land use change, taxation authorities need to update this information to determine appropriate tax value. They use integration in spatial data resources to local tax management systems, registries official data sources and castrates. This enables the authorities to have decision support systems to allow all local authorities to appropriately supervise taxation. Up-to-date of data on land prices with data on land use change that could help land use planning is important.

Table 4.1: Sample quality elements with weight for application

Application	APA$[m]$	CO [%]	CC [%]	LTC [%]	TV [%]	TCC [%]
Planning	1	4	5	4	3	80
	30	15	15	10	20	10
Taxation	0.5	1	4	4	3	75
	20	20	10	10	20	20

4.5 QUALITY REQUIREMENTS FOR APPLICATION

As we described in section 4.4, the requirement of quality evaluation for spatial data usage in taxation and planning depends on the intended use. To demonstrate the quality evaluation logic explained in section 3.5.3, we take into account some of quality elements by considering their importance to decide on dataset to use in the application. We assign arbitrary quality values corresponding to the quality elements as minimum values expected from the user as shown in table **4.1** to demonstrate the implementation.

- APA: Absolute positional accuracy
- CO: Completeness Omission
- CC: Completeness Commission
- TCC: Thematic Classification Correctness
- LTC: Logical Topological Consistency
- TV: Temporal Validity

As explained in section 3.5.3 the process of evaluating fitness for use of spatial dataset for application planning and taxation considers all specified quality elements. The process of checking the user quality requirements value is detailed in algorithm 1 section 3.4 .

4.6 DATA USED FOR PROTOTYPE IMPLEMENTATION

To implement the fitness for use reasoning logic based on application it is required to define theme_keywords as explained in chapter 3 section 3.5.2. Therefore, for demonstration purpose we used the following sample theme_keywords:

Planning :$\{Planning(0,5), landuse(0,1), cadastre(0,1), Parcel(0,1), Landcover(0,1)\}$

Taxation : $\{Taxation(0,5), Landuse(0,1), municipal(0,1), landownership(0,1), cadastralparcels(0,1)\}$
The application name and theme_keywords has assigned with their corresponding possible values during fitness for use evaluation. During searching for the application or theme_keyword matching, if similarity found in dataset overview quality or dataset description, it takes the corresponding maximum value. Otherwise if matching is not found it assign a zero value. For the application name the maximum value is determined based on the sum of the number of theme_keyword. In our example we define four synonyms for each of the application. Therefore, the application name found in the dataset has value higher than the sum of all synonyms. The values zero and

one assigned for the theme_keywords are used in similarity matching during fitness for use evaluation process. The fitness for use evaluation based on user application is explained in section 3.5.2.

In our use case we provided a of theme_keywords; however, there are more that could be used for an application. In a real scenarios more theme_keywords should be defined to get more search result. As explained in 3.5 the main idea is to get more number of spatial data resources as a recommendation candidate. Therefore, dataset found based on search by these predefined theme_keywords will be selected as useful dataset for further fitness for use evaluation process.

For the implementation purpose in a prototype system, we stored sample spatial datasets with arbitrary quality information in our database system as shown in table 4.2. We aimed to show how fitness for use evaluation can be used during searching spatial data resources based on user spatial data search requirements. However, in a real case, based on the logic we explained in section 3.4.3, spatial data resources should be extracted and populated into the system.

Table 4.2: Sample spatial data resources with quality information

Dataset name	APA$[m]$	CO$[\%]$	CC$[\%]$	TV $[\%]$	LTC $[\%]$	TCC $[\%]$
Land use	1.5	5	8	5	5	75
Land use dataset	1.5	5	8	5	5	75
Census Block map	0	3	6	3	7	84
Road line	0.5	1	3	0	5	100
LandCover	1	10	0	6	3	100
land form survey	2	0	0	3	7	95
Planning and cadastre	0.2	6	4	1	1	78
Plan for post office	0.5	1	3	0	5	100

4.7 SUMMARY

In this chapter we summarized the recommendation system design functional requirements and the case study used for the prototype implementation. The different functions given by the spatial data recommendation system are illustrated by use case model. The use-case is elaborated with detail description of each functional units. A summary of non functional requirement of the system is also presented. Furthermore, we describe the case study by introducing why we select the application domain taxation and planning. We proceeded to explain about the different quality requirements based on user intended use and associated spatial data needs. Following the explanation, we provide the selected quality elements used for the prototype implementation. Finally the data that we used for the prototype implementation and sample theme_keywords defined are explained. The implementation of the recommendation system based on the logic given in chapter 3 and based on the case study explained in this chapter is presented in the next chapter.

Chapter 5
Spatial data recommendation system in a prototype

5.1 INTRODUCTION

In this chapter a prototype implementation of the proposed spatial data recommendation system is presented. The implementation of the prototype is intended to realize the design of spatial data recommendation based on fitness for use evaluation logic presented in chapter 3. The selected case study to implement the designed system in a prototype was discussed in chapter 4. In the implementation we used PostgreSQL based Procedural Language/Structured Query Language(PL/pgLSQL) to create fitness for use evaluation functions to achieve the tasks inside the database. PHP:Hypertext Preprocessor Programming language used for web based user interface implementation. The prototype system is implemented using sample system default quality requirements with weight for given applications table 4.1 and sample spatial datasets given in table 4.2.

5.2 RECOMMENDATION SYSTEM USER INTERFACE

The recommendation system user interface implementation framework is shown in figure 5.1. The system allows users to specify their application requirements, spatial extent requirement using Openlayer map by drawing simple polygon, and the quality requirements with weight using input interface. Based on user specific requirements, the system search and display the results in the user output interface. The user output interface allow users to see and access the recommended datasets. Also it allows users to access the ranking functionality. These front end functionalities of the system are implemented using PHP and JavaScript.

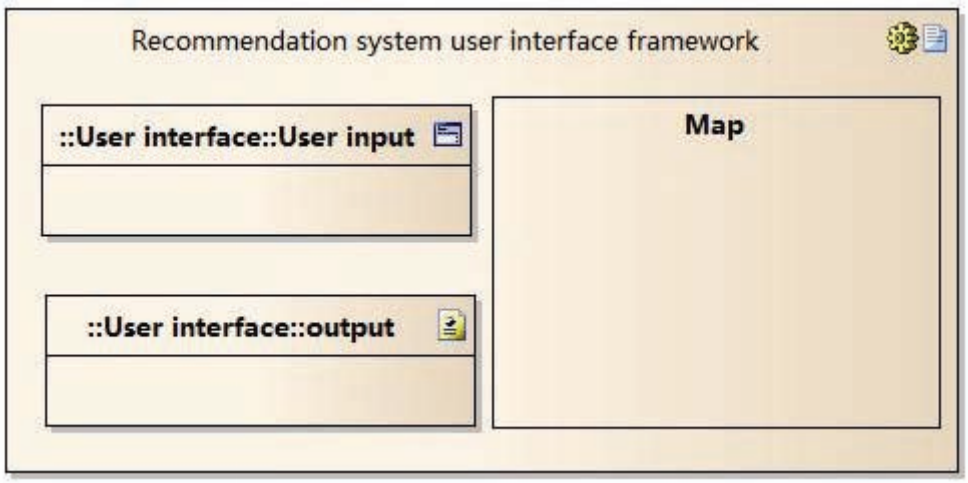

Figure 5.1: Web based user interface framework

5.3 PROFILE DATABASE IMPLEMENTATION

The back end of the spatial data recommendation system architecture introduced in section 3.2; which is used to store users requirements and spatial data quality information in a structured form is implemented to physical structure of system database. To implement spatial data recommendation system conceptual model we used Enterprise Architect system design tool which is Model Driven Architecture (MDA) technology [25]. The MDA is a systematic design approach for the development of software systems. The most important aspect of the MDA approach is the explicit identification of Platform-Independent Models (PIMs) which can be implemented on different platforms through Platform-Specific Models (PSMs). Then the PSM model will be transformed into actual code for the creation of the physical structure of the database.

Therefore, the conceptual data model given in section 3.3, which is a platform independent model (PIM), is transformed to a platform specific logical data model (PSM). Then the PSM transformed to physical model to generate PostgreSQL based data definition language (DDL) script. The DDL is used to create actual database of our system. PostgreSQL database management system is used as back end. The PL/pgSQL database programming language is used to develop fitness for use evaluation functionality and also in developing the prototype used inside the database.

5.4 RECOMMENDATION SERVICE

In the prototype implementation spatial data users can search spatial data resources by providing their application requirement, specifying quality requirements with corresponding weight, or both. Availability of such different options allow different users to use the system to search datasets based on their requirements. In the following subsections prototype implementation of the system functionalities that allows users to search spatial data based on their spatial data search quality requirements and access recommendation results as a system response will be discussed.

5.4.1 Registration/Login service

The spatial data recommendation system is based on user registration and login service. Therefore, to use the system users need to provide user name and password. Using the user unique information the system establish current user' session and allow spatial data search.

5.4.2 Recommendation system inputs

To generate recommendation based on fitness for use recommendation system requires users requirement and spatial data resources quality informations as input. Therefore, the system allows users to provide their spatial data search quality requirements. Figure 5.2 shows a screen shot of the recommendation system interface through which users can input their quality requirements. To search spatial data users can provide different inputs which are described below:

- **Extent information:** In order to search spatial data resource, system users require to specify their spatial extent requirement. As shown in figure 5.2 users can draw a polygon that refer their spatial extent requirement on the map and the system captures the extent. This extent information is mandatory input to proceed with searching spatial data based on fitness for use.

- **Quality requirement values:** The recommendation service evaluates datasets quality based on users spatial data search quality requirements as explained in section 3.5.3. The representation of quality values shown in figure 5.2 is percentage of error that can occur in the

datasets. The recommendation system has built in default quality values per application. The default quality values enable to search dataset that matches the minimum quality requirements for a specified application. However, users can modify the quality requirements and the system will make use of the modified quality requirements to search datasets that fits their requirements. The process of profiling user quality requirements is based on explanation and algorithm given in section 3.4.1.

- **Weight:** This value is required to enable users to give preference to their quality requirements. The system allows users to input the corresponding weight for each quality elements as shown on the user interface shown in figure 5.2. The assigned weight for all quality elements should be sum up to hundred. If users provide value for specific quality requirement without corresponding weight, the system marks incomplete input and forces users to provide weight. If users does not provide quality requirements and weight, the system default quality requirements with corresponding weight for application will be used to determine fitness for use of datasets.

- **Application:** Spatial data users can search datasets by specifying their application requirement. The application based quality information of spatial dataset can be obtained from data quality overview elements usage and purpose. The dataset description also serves to know about the dataset. In system prototype implementation, applications have predefined theme_keywords which are used in matching the overview quality information and dataset description. As shown on figure 5.2 users can choose the application for which they want a spatial datasets. The drop down box is populated by the application name defined by the system. When users an application, the system will identify the corresponding theme_keywords and the default quality value with corresponding weight which are defined during the system design. Using application and them_keywords for spatial data search allow to find more candidate spatial datasets that are made for different use but can serve the intended application as well.

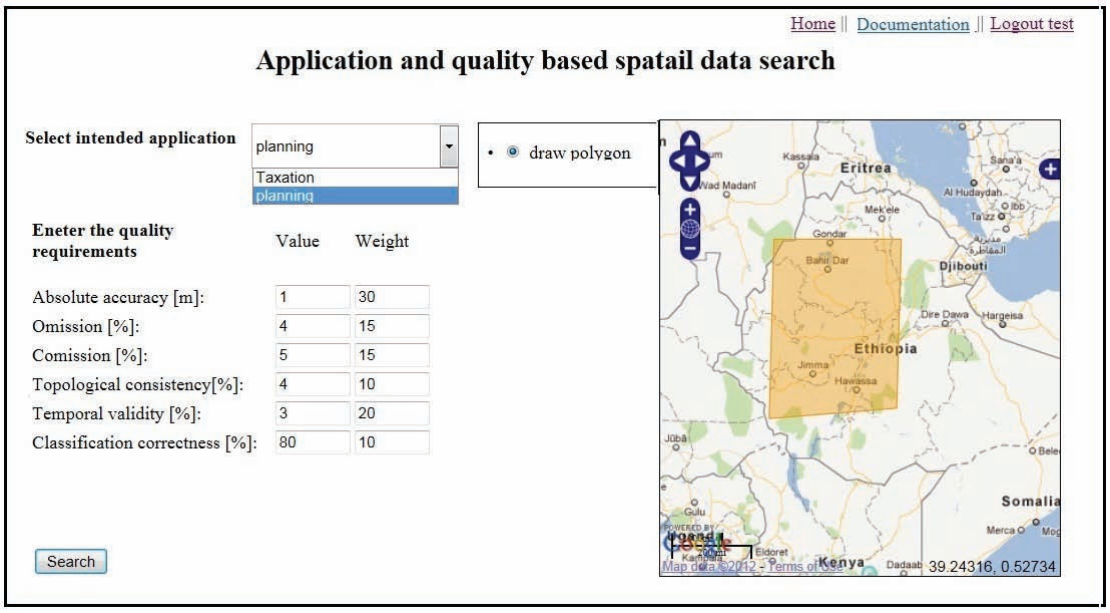

Figure 5.2: Spatial data recommendation user input interface

- **Spatial data resources**: It is an input in fitness for use based spatial data recommendation which could be extracted from the web and populated into the system as explained in section 3.4.3. For this prototype implementation we used the sample spatial data resources described in table 4.6.

5.4.3 Spatial dataset recommendation result

The recommendation service result page allows users to access recommended datasets as a response to their requirements. The recommendation process starts first by filtering spatial datasets based on user extent requirement. The datasets that satisfy user spatial extent requirement returned as a candidate datasets as explained in section 3.5.1. Then application based filtering process starts specifically by comparing the user application to the overview data quality information usage and purpose and continues matching the theme_keywords of the application to the description of datasets according to logic explained in section 3.5.2. Then the datasets that satisfies users extent and application requirements again evaluated based on users quality requirements with corresponding weight using the logic explained in section 3.5.3.

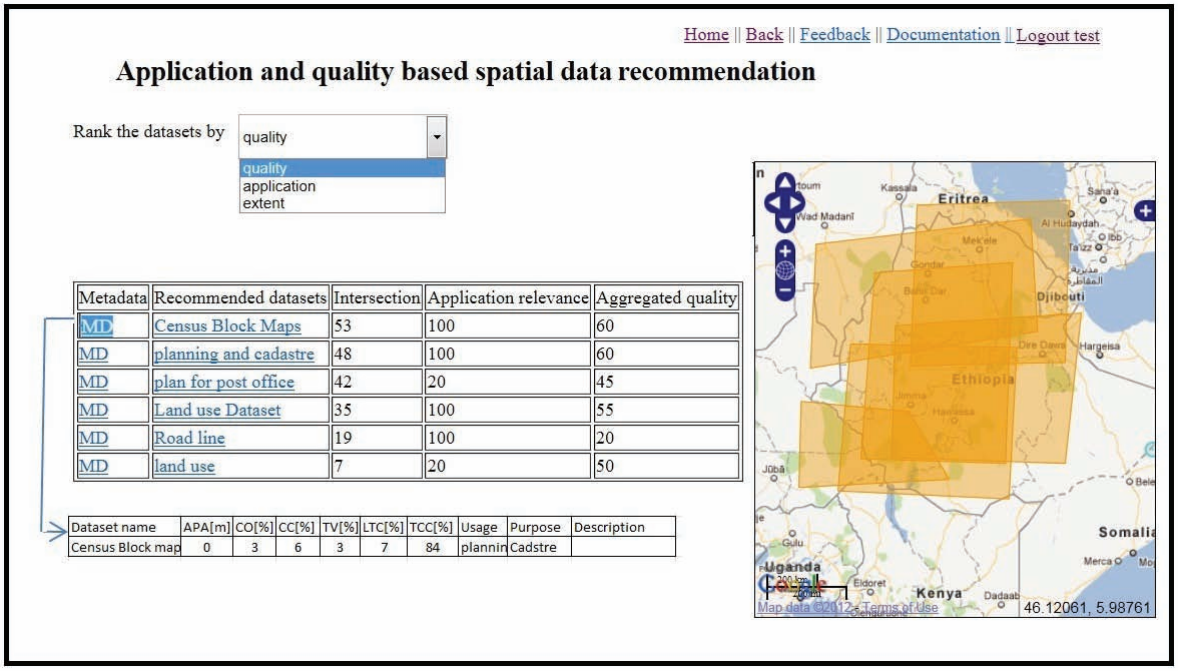

Figure 5.3: Recommended spatial datasets and metadata of selected dataset

Finally the system recommend spatial datasets with corresponding values that indicates level of relevance as computed by the fitness for use evaluation logic. The values returned with the recommended datasets enables users to easily observe which dataset best fits which requirements. In order to help users on picking the datasets based on extent, the computed spatial extent will also be visualized on the map. The visualization supports the recommendation functionality and enables users to visualize the spatial extent matching result.

5.4.4 Ranking service

The spatial data recommendation system recommends datasets based on user spatial data search quality requirements and provide a ranking service. The system returns the recommendation with default ranking orders: rank by extent, rank by application relevance and rank by quality relevance. However, the users can reorder the ranking criteria. Therefore, the system support the ranking service after recommendation. This ranking service enables users to change the ranking criteria to reorder the recommended datasets. This allows users to easily identify which dataset satisfy which requirements when there is long list recommended datasets.

A recommendation result for spatial data search with specific application and quality requirement given in figure 5.2 is displayed in figure 5.3. The result figure shows that the first two datasets have the same aggregated quality relevance which is based on ranking algorithm 18. In order to further identify the best datasets based on users quality preference, ranking mechanism by computing the distance of data quality element value from user requirements explained in algorithm 19 is used.

5.4.5 Dataset metadata view

Spatial data users sometimes may require to see the metadata of the recommended datasets, therefore based on their interest they can open the view option to display the metadata of datasets. This functionality allow users to explore quality informations of the datasets. Also if users want to explore on how the fitness for use evaluation generate the recommendation they can follow the documentation link.

5.4.6 System profile update

When spatial data users provide spatial data search quality requirements to search spatial data, the system update users' profile. The system captures users spatial data search quality requirements whenever users provide inputs and updates th user profile (UP). When users searches spatial data, most recent spatial data search quality requirements will be profiled in the system. Also spatial data resources profile (SDRP) get updated based on user interaction on the recommended datasets. Therefore, the update functionality is a unit that enable the system to generate appropriate recommendation for spatial data users based on their recent requirements.

5.4.7 System learn usability

Once the system generates recommendation datasets for users, during every interaction of user access to the datasets, the system tracks users access history without the users awareness. When users picked a dataset the system learn the event and update a counter. This count value incremented based on the picked event shows how many times a dataset has been visited by different users as explained in section 3.5.4. This information is used by the system to inform users about popularity of the datasets. The recommendation system make an update about spatial dataset access history according to the algorithms given in section 3.5.4.

Moreover, the recommendation system learn the actual spatial data usability information from users. After users surf the recommended datasets and make use of a dataset for their intended need, they are allowed to give feedback about dataset usability information using the explicit feedback mechanism. The users feedback helps the spatial data recommendation system to enhance fitness for use based recommendation. However, the dataset usability information explicitly provided from users require further analysis on how it can enhance the fitness for use evaluation reasoning logic. Users feedback mechanism can be implemented in different ways and different ways of interpretation can be devised. If users are allowed to give their comment in free form text a mechanism to interpret their feedback should be established.

5.5 SUMMARY

In this chapter a prototype implementation of spatial data recommendation service based on the selected case study has been presented. The recommendation system user interface framework and back end implantation of spatial data recommendation system profile database is described. The spatial data recommendation based on fitness for use includes different services that enable users to access the recommended datasets. The user interface design providing required spatial data search users inputs, displaying recommendation result page and other functionalities are also described. The prototype implementation result shows spatial data recommendation profiling based techniques and fitness for use evaluation logic simplify users effort to get best datasets for their intended use. The implementation of the fitness for use evaluation logic with some possible enhancements as recommended in chapter 6 can be used by spatial search engines to search spatial dataset based on fitness for use.

‘

Chapter 6
Discussion conclusion and recommendation

6.1 INTRODUCTION

This chapter summarizes the thesis work in recommending spatial datasets based on fitness for use. We discuss what is accomplished and draw conclusion in section 6.2 and finally wrap up this research work by indicating a possible direction for future work.

6.2 DISCUSSIONS AND CONCLUSIONS

The main objective of this research has been designing a reasoning logic to determine fitness for use of a spatial datasets, and then to use the fitness for use as a search criteria to search and recommend spatial dataset that fit users intended use. To achieve this objective we reviewed literatures about spatial data quality, concept of fitness for use and recommendation technology. Given this background we are able to propose a spatial data recommendation system data model and profiling, fitness for use reasoning logic design, and implementation of the logic in a prototype to recommend spatial dataset for users. We used selected case study as a proof of concept.

The fundamental approach to determine fitness for use is comparison of users quality requirements and quality of data resources. In order to use fitness for use as a searching criteria in GIS, comprehensive comparison against user quality requirement and detailed quality description of spatial dataset is required. Thus understanding users' view towards spatial data quality and quality description of spatial data resources, gave us an idea on how to design a concptual model of recommendation system presented in figure 3.2. As discussed in chapter 2, in GIS spatial data quality is a perception or an assessment of data fitness to serve its purpose in a given context and subjective to various applications. Widely accepted expression affirms that spatial data quality is recognized in terms of its specific use and the quality definition given by ISO is accepted in common to describe spatial data quality.

Therefore, we followed the spatial data quality according to ISO standard to represent the spatial data quality and users spatial data search quality requirements in our system. This simplification is required because the standard gives common ground on spatial data quality to evaluate its fitness for use. We also consider OGC catalogue service as a source to extract required datasets quality description. However, data quality description to determine fitness for use should not be limited to the ISO standard. Other factors such as: currency, cost, accessibility of the dataset, dataset granularity, popularity and users opinion about the dataset need to be included.

The recommendation system data model is used to profile user spatial data search quality requirements in UP, selected spatial data quality descriptions in SDRP, dataset usability, and user data access history in IP. In this recommendation system for fitness for use evaluation extent of dataset, quantitative and qualitative quality description of the datasets and other informations are profiled in SDRP. The recommendation system tracks users data access history to build dataset popularity information. Moreover, it allows users to provide usability feedback on their usage of a dataset. This indirectly gathered information about spatial datasets usability enhance the

fitness for use evaluation logic in recommending spatial datasets that best fits users requirements. The system data model is also flexible to add other quality elements to extend the fitness for use evaluation with some adjustment.

Given the data model, in order to represent, initialize and update profile we further discussed selected profiling techniques and designed algorithms that describe profiling procedures. After exploring on techniques and procedures to used in profiling to structure the required information for spatial data recommendation in the system, we designed fitness for use evaluation logic. The designed fitness for use evaluation logic includes extent matching, application name matching, and quantitative quality evaluation as explained in section 3.5. The techniques used in the logic design includes matching, comparison and filtering techniques. In the process of fitness for use evaluation, these techniques facilitate to search for the best available spatial data resources.

The fitness for use evaluation logic in the system uses user spatial data searching quality requirements as a criteria to search datasets and recommend suitable data for users. If users do not provide specific quality requirements, the system search best dataset based on user application relevance, extent requirement and based on dataset popularity. This means that even though spatial datasets are not provided with specific quality description by users as well as data producers, fitness for use can be determined using spatial data recommendation profiling to enable users to get the dataset that fits their application. Therefore, a spatial data recommendation profiling that contains detail users quality requirements and information about usability of datasets is advantageous. Moreover, spatial data recommendation profiling is helpful not only to determine fitness for use of datasets, but also spatial data producers can benefit using the system as a knowledge sources about their datasets.

Currently web technology provides a great contribution to access spatial data resources. To allow users to access spatial data resources in easy way the OpenSearch-Geo extensions standard provide mechanism to query a resources based on geographic extent or location name. This advance the process of searching spatial data resources on the web environment. Even though the OpenSearch-Geo extensions help to search spatial resources, it is up to the users to choose the best spatial data resources for their intended use by looking into the metadata. Furthermore, various type of spatial data becomes available and shared on the web. Hence it became more and more difficult for users to choose the best data from the search results. Therefore, in addition to allowing standardized query for spatial data search using extent and location name, a mechanism to search spatial data resources based on users spatial data search quality requirement is beneficial. As one possible solution fitness for use evaluation reasoning logic proposed in this thesis work can be used as a search criteria to search spatial datasets based on user spatial data search quality requirements.

Current spatial data search engines for searching spatial data available on SDI are limited to searching by spatial extent and geographical location name and return metadata. In this aspect, the designed reasoning logic by profiling users spatial data search quality requirements and data quality description is valuable to recommend best data resources for users as per their requirements. Hence, spatial data recommendation profiling approach used to determine fitness for use can advance the current spatial data search engine to search data based on fitness for use. In order to integrate the fitness for use evaluation logic in current spatial data search engines, perhaps implementation modification is required.

In conclusion, we attained the stated objectives by demonstrating the fitness for use evaluation logic with structured set of spatial data search quality requirements which is implemented on a prototype based on the selected case study. The implementation result from prototype shows that spatial data recommendation profiling to determine fitness for use of spatial datasets is an essential component. Also the proposed fitness for use reasoning logic to search spatial data is beneficial

and implementable to integrate into the current spatial data search engine with some modification of logic implementation.

6.3 RECOMMENDATIONS

To enhance the proposed spatial data recommendation based on fitness for use evaluation reasoning logic and its use to recommend spatial data resources, we suggest the following research directions:

1. In designing fitness for use evaluation logic for spatial data recommendation we covered spatial extent, overview quality description, quantitative quality description of the dataset, and datasets popularity. However, lineage, currency, and actual dataset usability feedback from users enhance the reasoning logic to determine fitness for use of spatial datasets. Hence, for further study we recommend the spatial data recommendation profiling and fitness for use evaluation reasoning to include these information.

2. Due to time limitation, we implemented and tested fitness for use reasoning logic using static data populated in the system database manually with arbitrary quality information. However, the system has to search for spatial datasets from the web and extract all the required data quality descriptions to build the spatial data resource profile based on user spatial data search quality requirements. We suggest further research on extraction of spatial data resources from the web and populate into spatial data resource profile (SDRP). To achieve this detail exploration on ISO model content and structure of metadata, data repository and web infrastructures is required. In addition, we recommend further research on enhancing the use of fitness for use reasoning logic by integrating it into current geoportals.

3. Due to complexity of the research problem we design the fitness for use reasoning logic based on ISO standard spatial data quality description. We suggest further research work on how to evaluates a spatial dataset if its quality description is not presented according to the ISO standard.

4. We also would like to suggest future research to enhance the fitness for use evaluation logic to recommend partial spatial data resources and derived spatial data resources using web processing services.

LIST OF REFERENCES

[1] PL/pgSQL - SQL Procedural Language. `http://www.postgresql.org/docs/8.3/static/plpgsql.html`, 2011.

[2] E. and B. Vaßeur. How to select the best dataset for a task. In *Proceedings of 3rd International Symposium on Spatial Data Quality (ISSDQ'04)*, pages 197–206, 2004.

[3] ISO/TC 211. Text of 19113 geographic information - quality principles, as sent to the iso central secretariat for registration as fdis, 2002.

[4] A. Agumya and G.J. Hunter. A risk-based approach to assessing the fitness for use of spatial data. *URISA Journal*, 11(1):33–44, 1999.

[5] M.P. Armstrong, G. Rushton, J. Chakraborty, A.W. Ibaugh, and A.J. Ruggles. Spatial data systems for transportation planning. 1997.

[6] R. Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, 2002.

[7] R. Burke. Hybrid web recommender systems. In *The adaptive web*, pages 377–408. Springer-Verlag, 2007.

[8] M. Caprioli, A. Scognamiglio, G. Strisciuglio, and E. Tarantino. Rules and standards for spatial data quality in gis environments. In *INTERNATIONAL CARTOGRAPHIC CONFERENCE–ICC*, volume 21, pages 10–16, 2003.

[9] A.D. Chapman and Global Biodiversity Information Facility. *Principles of data quality*. Global Biodiversity Information Facility, 2005.

[10] European Commission et al. Draft implementing rules for metadata (version 3), 2007.

[11] A. Coote and L. Rackham. Neogeographic data quality–is it an issue. In *Annual Conference of the Association for Geographic Information, AGI*, 2008.

[12] Victorian Spatial Council. Spatial Information Data Quality Guidelines. Victorian Spatial Council, 2009.

[13] W.H. Delone and E.R. McLean. The delone and mclean model of information systems success: A ten-year update. *Journal of management information systems*, 19(4):9–30, 2003.

[14] R. Devillers, Y. Bédard, R. Jeansoulin, and B. Moulin. Towards spatial data quality information analysis tools for experts assessing the fitness for use of spatial data. *International Journal of Geographical Information Science*, 21(3):261–282, 2007.

[15] R. Devillers, M. Gervais, Y. Bédard, and R. Jeansoulin. Spatial data quality: from metadata to quality indicators and contextual end-user manual. In *OEEPE/ISPRS Joint Workshop on Spatial Data Quality Management*, pages 21–22, 2002.

[16] R. Devillers and R. Jeansoulin. *Fundamentals of spatial data quality*. Wiley Online Library, 2006.

[17] R. Devillers, A. Stein, Y. Bédard, N. Chrisman, P. Fisher, and W. Shi. Thirty years of research on spatial data quality: Achievements, failures, and opportunities. *Transactions in GIS*, 14(4):387–400, 2010.

[18] E. Fekpe. Spatial Data Quality and Transportation Applications. In *TS23.6, Promoting Land Administration and Good Governance*. 5th FIG Regional Conference, Accra, Ghana„ 2006.

[19] O. Fonts, J. Huerta, L. Díaz, and C. Granell. OpenSearch-Geo: The simple standard for geographic web search engines, 2010.

[20] A.U. Frank. Metamodels for data quality description. *Data quality in Geographic Information: From error to uncertainty*, pages 15–29, 1998.

[21] A.U. Frank, E. Grum, and B. Vaßeur. Procedure to select the best dataset for a task. *Geographic Information Science*, pages 81–93, 2004.

[22] M.A. Gebresilassie, I. Ivánová, and J. Morales. User profiles for data quality models. Master's thesis, University of Twente Faculty of Geo-Information and Earth Observation ITC, 2011.

[23] J. Griffith, C. O'Riordan, and H. Sorensen. Identifying and analyzing user model information from collaborative filtering datasets. *Personalization techniques and recommender systems*, 70:165, 2008.

[24] K.T. Huang, Y.W. Lee, and R.Y. Wang. *Quality information and knowledge*, volume 141. Prentice Hall PTR, 1999.

[25] ing. Jan van Bennekom-Minnema. The Land Administration Domain Model 'Survey Package' and Model Driven Architecture. `http://www.janvanbennekom.nl/mscthesis.html`, October.

[26] ISOTC211. Revised text of 19115 Geographic information - Metadata, as sent to the ISO Central Secretariat for registration as FDIS , 2003.

[27] ISOTC211. Text of 19114 geographic information - quality evaluation procedures as sent to the iso central secretariat for publication, 2003.

[28] I. Ivánová, J. Morales, M. A. Gebresilassie, and R. A. de By. Searching for spatial data resources by fitness for use, 2011.

[29] M. Jahn. User needs in a maslow schemata. In *Proceedings of the International Symposium on Spatial Data Quality 2004*, pages 169–182, 2004.

[30] D.J. Maguire and P.A. Longley. The emergence of geoportals and their role in spatial data infrastructures. *Computers, Environment and Urban Systems*, 29(1):3–14, 2005.

[31] M. Maguire and N. Bevan. User requirements analysis. In *Proceedings of IFIP 17th World Computer Congress*, pages 133–148, 2002.

[32] S.E. Middleton, D.C. De Roure, and N.R. Shadbolt. Capturing knowledge of user preferences: ontologies in recommender systems. In *Proceedings of the 1st international conference on Knowledge capture*, pages 100–107. ACM, 2001.

[33] S. M. Z. S Mohamed Ghouse. Modeling spatial variation of data quality in databases. Master's thesis.

[34] M. Montaner, B. López, and J.L. De La Rosa. A taxonomy of recommender agents on the Internet. *Artificial intelligence review*, 19(4):285–330, 2003.

[35] G. Navratil, J. Scholz, L. Danek, and F. Karimipour. Data for spatial planning–a comparison of three cities.

[36] D. Nebert, A. Whiteside, and P.P. Vretanos. OpenGIS catalogue services specification 2.0. 2-ISO. *Open Geospatial Consortium Inc*, 2007.

[37] J. Nogueras-Iso, FJ Zarazaga-Soria, J. Lacasta, R. Béjar, and PR Muro-Medrano. Metadata standard interoperability: application in the geographic information domain. *Computers, environment and urban systems*, 28(6):611–634, 2004.

[38] P.A.J. Oort. Spatial data quality: from description to application. 2006.

[39] T.C. Redman. *Data quality for the information age*. Artech House, 1996.

[40] T.C. Redman. *Data quality: the field guide*. Digital Pr, 2000.

[41] EEA Technical report. Land in Europe: prices, taxes and use patterns. Europian Environmental Agency, 2010.

[42] S. Rizzi, A. Abelló, J. Lechtenbörger, and J. Trujillo. Research in data warehouse modeling and design: dead or alive? In *Proceedings of the 9th ACM international workshop on Data warehousing and OLAP*, pages 3–10. ACM, 2006.

[43] S. Schiaffino and A. Amandi. Intelligent user profiling. In *Artificial intelligence*, pages 193–216. Springer-Verlag, 2009.

[44] K. Senkler, U. Voges, and A. Remke. An iso 19115/19119 profile for ogc catalogue services csw 2.0. In *Workshop paper presented at 10th EC-GI & GIS Workshop, Warsaw, Poland, June*, pages 23–25, 2004.

[45] Y.L. Simmhan, B. Plale, and D. Gannon. Towards a quality model for effective data selection in collaboratories. 2006.

[46] X. Song, X. Rui, W. Hou, and H. Tan. An OGC standard-oriented architecture for distributed coal mine map services. *JOURNAL OF CHINA UNIVERSITY OF MINING & TECHNOLOGY*, 18(3):381–385., 2008.

[47] K. Swearingen and R. Sinha. Interaction design for recommender systems. In *Designing Interactive Systems*, volume 6, pages 312–334. Citeseer, 2002.

[48] G.K. Tayi and D.P. Ballou. Examining data quality. *Communications of the ACM*, 41(2):54–57, 1998.

[49] A. Turner. OpenSearch-Geo Extensions. `http://www.opensearch.org/Specifications/OpenSearch/Extensions`, August 2007.

[50] A. Zargar and R. Devillers. An operation-based communication of spatial data quality. In *2009 International Conference on Advanced Geographic Information Systems & Web Services*, pages 140–145. IEEE, 2009.

[51] P. Zhao, A. Chen, Y. Liu, L. Di, W. Yang, and P. Li. Grid metadata catalog service-based ogc web registry service. In *Proceedings of the 12th annual ACM international workshop on Geographic information systems*, pages 22–30. ACM, 2004.

Appendix A
Activity diagram

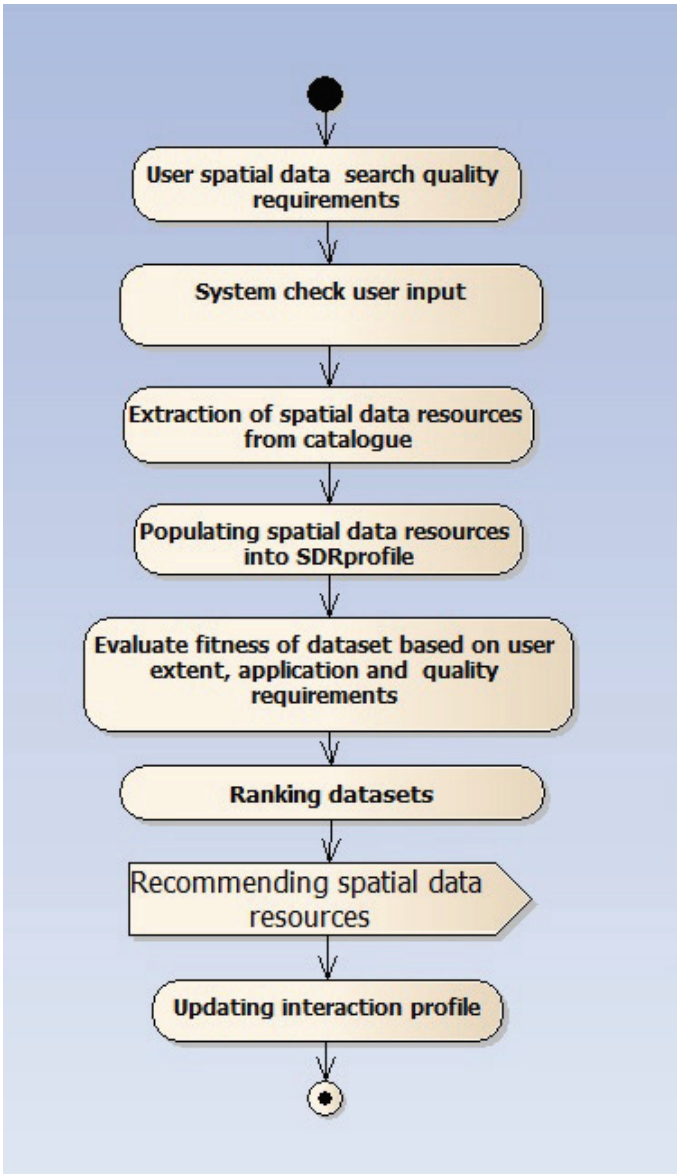

Figure A.1: Spatial data recommendation process

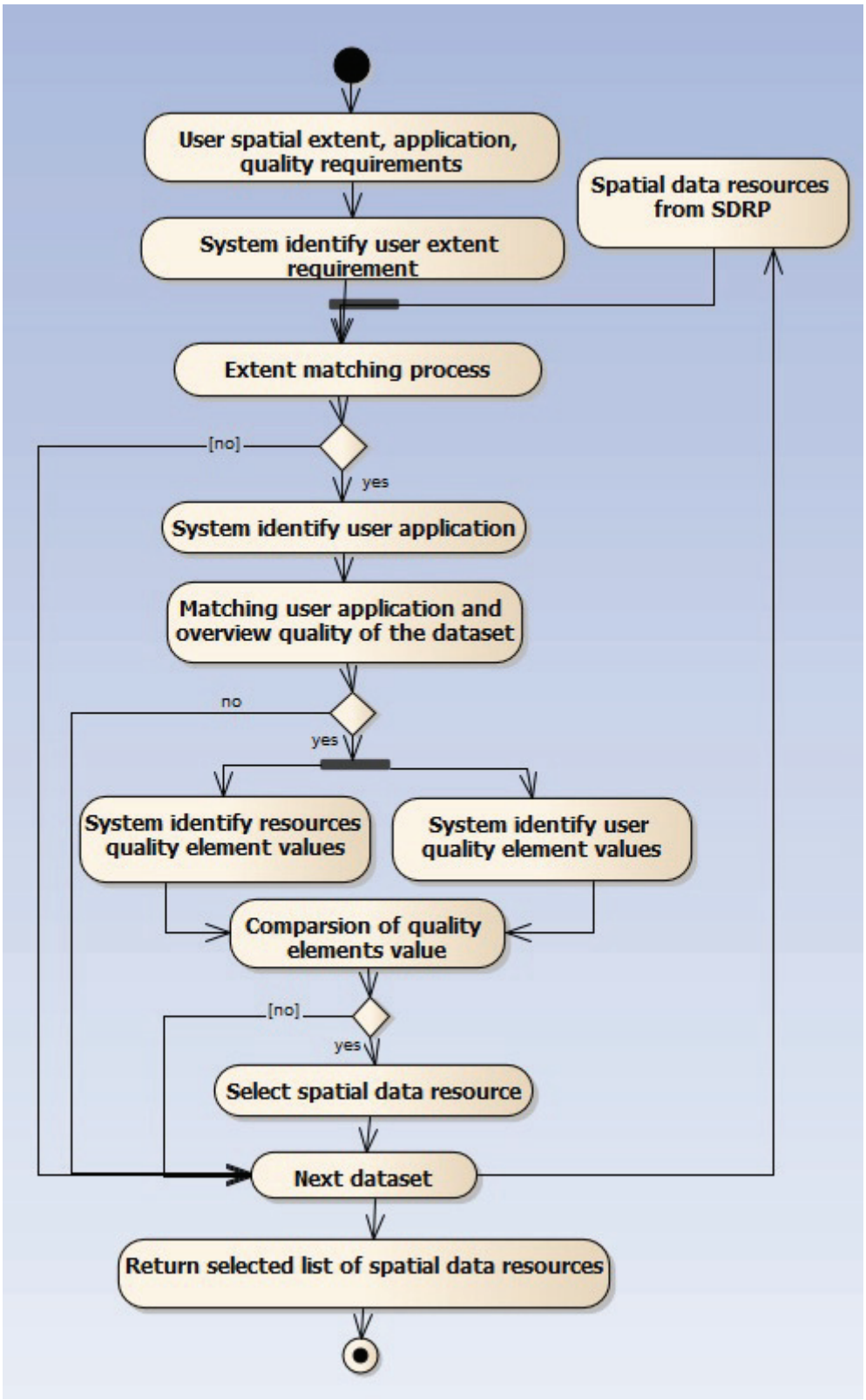

Figure A.2: Spatial data fitness for use evaluation activity

Appendix B

Fitness for use evaluation functions

In this section implementation of fitness for use evaluation algorithm is given. The program is written in PL/pgSQL database programming language. The functions defined here are used in other parts of the system programs for evaluating fitness of a datasets based on user requirements.

```
//=============================================================
// Spatial extent matching -- Algorithm 4,5
CREATE OR REPLACE FUNCTION extent_matching(integer)
RETURNS SETOF tbltemp_e AS
$BODY$
DECLARE
ex_id ALIAS FOR $1;
r tbltemp_e;
Intersection_area tbltemp_e;
percentageofinterseperDS tbltemp_e;
ue   e_x__bounding_polygon%rowtype ;
BEGIN
SELECT * INTO ue
FROM e_x__bounding_polygon
WHERE ex_boundingpolygon_id = ex_id;// fetch user extent information by id
FOR r IN
        SELECT sdrprofile_id, data_set_u_r_i ,ex1.ex_boundingpolygon_id ,
                ex1.the_geom
        FROM e_x__bounding_polygon ex1 ,  s_d_r_profile, intersection_has
        WHERE intersection_has.e_x__bounding_polygon_ex_boundingpolygon_id =
        ex1.ex_boundingpolygon_id AND s_d_r_profile.sdrprofile_id =
        intersection_has.s_d_r_profile_sdrprofile_id AND
        ex1.ex_boundingpolygon_id <> ue.ex_boundingpolygon_id
LOOP
IF ST_WITHIN(ST_centroid(ue.the_geom), r.the_geom) THEN
        r.Intersection_area =
                ST_GeomFromEWKT(ST_Intersection(ue.the_geom, r.the_geom));
        r.percentageOfIntrse=
                 (ST_Area(r.Intersection_area)/ST_Area(ue.the_geom))*100;
        r.percentageofinterseperDS=
                (ST_Area(r.Intersection_area)/ST_Area(r.the_geom))*100;
RETURN NEXT r;
ELSE
IF ST_INTERSECTs(ue.the_geom, r.the_geom)  THEN
        r.Intersection_area =
                ST_GeomFromEWKT(ST_Intersection(ue.the_geom, r.the_geom));
        r.percentageOfIntrse=
                (ST_Area(r.Intersection_area)/ST_Area(ue.the_geom))*100;
        r.percentageofinterseperDS=
                (ST_Area(r.Intersection_area)/ST_Area(r.the_geom))*100;
RETURN NEXT r;
ELSE
IF  ST_WITHIN(ST_centroid(ue.the_geom), r.the_geom) THEN
```

```
r.Intersection_area =
        ST_GeomFromEWKT(ST_Intersection(ue.the_geom, r.the_geom));
r.percentageOfIntrse=
        (ST_Area(r.Intersection_area)/ST_Area(ue.the_geom))*100;
r.percentageofinterseperDS=
        (ST_Area(r.Intersection_area)/ST_Area(r.the_geom))*100;
RETURN NEXT r;
END IF;
END IF;
END IF;
END LOOP;
RETURN ;
END;
$BODY$
LANGUAGE PLPGSQL VOLATILE

//=============================================================
//application based and theme_keywords matching
// Algorithm 8

CREATE OR REPLACE FUNCTION extent_appliction(INTEGER,CHARACTER)
RETURNS SETOF tbltempa AS
$BODY$
DECLARE
r application%rowtype;
my_row tbltempa%rowtype;
usrext alias FOR $1;
app alias FOR $2;
BEGIN
IF app = 'planning' THEN
FOR my_row IN
        SELECT DISTINCT ON (qe.sdrprofile_id) qe.sdrprofile_id,
                qe.data_set_u_r_i, usage_1, purpose_1,
                qe.percentageofintrse ,
                qe.percentageofinterseperds ,
                datasetname,
                st_astext (ST_GeomFromWKB (qe.the_geom,900913))
        FROM  s_d_r_profile sdr,extent_matching(usrext) qe,
                 overview_quality_elements
        WHERE sdr.overview_quality_elements_overviewqualityelements_id
                 = overview_quality_elements.overviewqualityelements_id
                 AND sdr.sdrprofile_id= qe.sdrprofile_id
                 AND (overview_quality_elements.purpose_1 ~ 'land cover' OR
                       overview_quality_elements.usage_1 ~ 'land cover' OR
                       overview_quality_elements.purpose_1 ~ 'Land-Use' OR
                       overview_quality_elements.usage_1 ~ 'Land-Use' OR
                       overview_quality_elements.purpose_1 ~ 'cadastre' OR
                       overview_quality_elements.usage_1 ~ 'cadastre' OR
                       overview_quality_elements.purpose_1 ~ 'Taxation' OR
                       overview_quality_elements.usage_1 ~ 'Taxation' OR
                       overview_quality_elements.purpose_1 ~ 'land cover' OR
                       overview_quality_elements.usage_1 ~ 'land cover')
LOOP
 RETURN NEXT my_row;
END LOOP;
END IF;
 RETURN;
END
$BODY$
LANGUAGE 'plpgsql';
```

```
//-------------------------------------------------------------------------

//Quantify application name and theme_keywords found in the dataset ovq
//Algorithm 9 -10

CREATE OR REPLACE FUNCTION  extent_appliction_weight(INTEGER,CHARACTER)
RETURNS SETOF tbltempa AS
$BODY$
DECLARE
my_row tbltempa%rowtype;
usrext alias FOR  $1;
app ALIAS FOR $2;
BEGIN
FOR my_row IN Select  * FROM extent_appliction0(usrext,app)
LOOP
IF my_row.usage_1 ~ 'planning' or my_row.purpose_1 ~ 'planning' THEN
 my_row.w_application =  5;
 ELSE
 my_row.w_application = 0;
 END IF;
IF my_row.usage_1 ~ 'Land-Use' or my_row.purpose_1 ~ 'Land-Use' THEN
 my_row.w_land_use= 1;
 ELSE
 my_row.w_land_use= 0;
 END IF;
IF my_row.usage_1 ~ 'cadstre' or my_row.purpose_1 ~ 'cadstre' THEN
 my_row.w_cadastre= 1;
 ELSE
 my_row.w_cadastre= 0;
 END IF;
IF my_row.usage_1 ~ 'Taxation' or  my_row.purpose_1 ~ 'Taxation'  THEN
 my_row.w_Taxation= 1;
 ELSE
 my_row.w_Taxation= 0;
 END IF;
IF my_row.usage_1 ~ 'land cover' or my_row.purpose_1 ~ 'land cover' THEN
 my_row.  w_Land_cover = 1;
 ELSE
 my_row.  w_Land_cover = 0;
 END IF;
 my_row.t_tkw_weight=my_row.w_land_use + my_row.w_cadastre
        + my_row.w_Taxation +  my_row.  w_Land_cover;
RETURN NEXT my_row;
END LOOP;
RETURN;
END
$BODY$
LANGUAGE 'plpgsql';
//-------------------------------------------------------------------------
//Display relevance of datasets based on application and theme_keywords weight
//Algorithm 11
CREATE OR REPLACE FUNCTION extent_appliction_rec(INTEGER,CHARACTER)
RETURNS SETOF tbltempa AS
$BODY$
DECLARE
my_row tbltempa%rowtype;
userext ALIAS FOR $1;
app ALIAS FOR $2;
BEGIN
```

```
FOR my_row IN SELECT  * FROM extent_appliction_weight0(userext,app)
LOOP
If my_row.w_application = 5 THEN
my_row.relevance = 100;
ELSIF my_row.t_tkw_weight = 4 THEN
my_row.relevance = 80;
ELSIF my_row.t_tkw_weight =3 THEN
my_row.relevance = 60;
ELSIF my_row.t_tkw_weight=2 THEN
my_row.relevance = 40 ;
ELSIF my_row.t_tkw_weight = 1 THEN
my_row.relevance = 20;
ELSE
my_row.relevance = 0;
END IF;
my_row.area_ratio_per_user = round(my_row.percentageofintrse);
my_row.area_ratio_per_ds = round(my_row.percentageofinterseperds);
RETURN NEXT my_row;
END LOOP;

RETURN;
END
$BODY$
LANGUAGE 'plpgsql';

//=============================================================
//Fitness for use evaluation by quality
//Compute user quality requirement range -- Algorithm 13
CREATE OR REPLACE FUNCTION user_quality_range(INTEGER)
RETURNS SETOF tbltemp_4j AS
$BODY$
DECLARE
r d_q__sub_element%rowtype;
my_row tbltemp_4j%rowtype;
userid ALIAS FOR $1;
begin
SELECT INTO my_row  * FROM d_q__sub_element
WHERE d_q__sub_element.user_profile_userprofile_id = userid;
IF (my_row.d_q__abso_exter_posi_accur = 0 ) THEN
my_row.abmin = -0.5;
my_row.abmax = 0.5;
ELSE
my_row.abmin =
-(my_row.d_q__abso_exter_posi_accur)-( my_row.d_q__abso_exter_posi_accur/2);
my_row.abmax =
my_row.d_q__abso_exter_posi_accur + ( my_row.d_q__abso_exter_posi_accur/2);
END IF;
my_row.omin =   my_row. d_q__comp_omission  - ( my_row. d_q_comp_omission /2);
my_row.omax =   my_row. d_q__comp_omission  + ( my_row. d_q_comp_omission /2);
my_row.cmini =  my_row. d_q__com_commission  - ( my_row. d_q_com_commission /2);
my_row.cmaxi = my_row. d_q__com_commission  + ( my_row. d_q_com_commission /2);
my_row.tclmin = my_row.d_q__them_cassif_correct - (my_row.d_q_them_cassif_correct /2);
my_row.tclmax = my_row.d_q__them_cassif_correct + (my_row.d_q_them_cassif_correct /2);
my_row.tmpvmin= my_row. dq__temp_validity - (my_row. dq_temp_validity /2);
my_row.tmpvmax= my_row. dq__temp_validity + (my_row. dq_temp_validity /2);
my_row.ltoconmin =my_row.d_q__logic_topo_consis - (my_row.d_q_logic_topo_consis /2);
my_row.ltoconmax = my_row.d_q__logic_topo_consis + (my_row.d_q_logic_topo_consis /2);
RETURN NEXT my_row;
END
$BODY$
```

```
LANGUAGE 'PLPGSQL';

//-------------------------------------------------------------------------------------------

//Evaluation of data quality of dataset based on user quality range
//datasets based on user range -- Algorithm 14
CREATE OR REPLACE FUNCTION evaluate_quality_in_range(INTEGER,INTEGER,CHARACTER)
RETURNS SETOF tbltemp_eaq AS
$BODY$
DECLARE
r tbltemp_4j%rowtype;
my_row tbltemp_eaq%rowtype;
userid alias FOR $1;
uext ALIAS FOR $2;
appn ALIAS FOR $3;
BEGIN
SELECT INTO r * from user_quality_range(userid) ;-- call a function
FOR my_row IN Select DISTINCT ON ( sdrprofile_id) sdr.sdrprofile_id,
sdr.data_set_u_r_i, dq.d_q__abso_exter_posi_accur ,dq.d_q__com_commission ,
dq.d_q__comp_omission , dq.d_q__logic_topo_consis, dq.d_q__temp_consis,
dq.dq__temp_validity, dq.d_q__them_cassif_correct, dq.user_profile_userprofile_id,
dq.wt__d_q__sub_element_wt_dq_subelement_id,percentageofinterseperds,
percentageofintrse,sdr.datasetname, dse.relevance
FROM (s_d_r_profile AS sdr INNER JOIN extent_appliction_relevance(uext,appn)
AS dse ON sdr.sdrprofile_id = dse.sdr_id) INNER JOIN d_q__sub_element AS
dq ON sdr.d_q__sub_element_dq_subelement_id = dq.dq_subelement_id AND
dq.user_profile_userprofile_id IS NULL
WHERE dq.d_q__abso_exter_posi_accur >= r.abmin AND
dq.d_q__abso_exter_posi_accur <= r.abmax OR
dq.d_q__comp_omission >= r.omin AND
dq.d_q__comp_omission <= r.omax OR dq.d_q__com_commission >= r.Cmini
AND dq.d_q__com_commission <= r.Cmaxi OR dq.d_q__them_cassif_correct >= r.tclmin
AND dq.d_q__them_cassif_correct <= r.tclmax OR dq.dq__temp_validity >= r.tmpvmin
AND dq.dq__temp_validity <= r.tmpvmax OR dq.d_q__logic_topo_consis >= r.ltoconmin
AND dq.d_q__logic_topo_consis <= r.ltoconmax
LOOP
RETURN NEXT my_row;
END LOOP;
RETURN;
END
$BODY$
LANGUAGE 'plpgsql';
//-------------------------------------------------------------------------------------------
//Quantify the dataset data quality subelements based on user quality subelements
//Algorithm 15-16
CREATE OR REPLACE FUNCTION
weighted_ds_quality_based_on_user_weight(INTEGER,INTEGER,ChARACTER)
RETURNS SETOF tbltemp_eaq AS -- Algorithm 14 - 15,
$BODY$
DECLARE
r d_q__sub_element%rowtype;
my_row tbltemp_eaq%rowtype;
ur d_q__sub_element%rowtype;
rng tbltemp_4j%rowtype;
userid ALIAS FOR $1;
uext ALIAS FOR $2;
appn ALIAS FOR $3;
urw wt__d_q__sub_element%rowtype;
BEGIN
SELECT INTO urw *
```

```
FROM  wt__d_q__sub_element , d_q__sub_element AS dq, user_profile
WHERE wt__d_q__sub_element_wt_dq_subelement_id = wt_dq_subelement_id
          AND dq.user_profile_userprofile_id = userid;
SELECT INTO rng * from user_quality_range(userid);
FOR my_row IN SELECT * FROM  evaluate_quality_in_range(userid,uext,appn)
LOOP
IF my_row.d_q__abso_exter_posi_accur >= rng.abmin
AND my_row.d_q__abso_exter_posi_accur<=
rng.abmax THEN
my_row.b_absolute_posti = 1 * urw.wt_d_q__abso_exter_posi_accurme;
ELSE
my_row. b_absolute_posti = 0 * urw.wt_d_q__abso_exter_posi_accurme;
END IF;
IF my_row.d_q__com_commission >= rng.Cmini AND
 my_row.d_q__com_commission <=rng.Cmaxi THEN
my_row.b_commission = 1 * urw.wt_d_q__com_commission ;
ELSE
my_row.b_commission = 0 * urw.wt_d_q__com_commission;
END IF;
IF my_row.d_q__comp_omission >= rng.omin AND
my_row.d_q__comp_omission<=rng.omax THEN
my_row. b_omission = 1 * urw.wt_d_q__com_commission;
ELSE
my_row. b_omission = 0 * urw.wt_d_q__com_commission;
END IF;
IF my_row.d_q__them_cassif_correct>= rng.tclmin AND
 my_row.d_q__them_cassif_correct<=rng.tclmax THEN
my_row. b_theme_classification = 1 * urw.wt_d_q__them_cassif_correct ;
ELSE
my_row. b_theme_classification = 0 * urw.wt_d_q__them_cassif_correct ;
END IF;
IF my_row. dq__temp_validity>= rng.tmpvmin AND
my_row. dq__temp_validity <= rng.tmpvmax THEN
my_row.b_temporal_validity = 1 * urw.  wt_d_q__temp_validity;
ELSE
my_row.b_temporal_validity = 0 * urw. wt_d_q__temp_validity;
END IF;
IF my_row. d_q__logic_topo_consis >= rng.ltoconmin AND
my_row.d_q__logic_topo_consis <= rng.ltoconmax THEN
my_row.b_topological_consi  = 1 * urw. wt_d_q__logic_topo_consis;
ELSE
my_row.b_topological_consi = 0 * urw. wt_d_q__logic_topo_consis;
END IF;
my_row.weight_agg = my_row.b_absolute_posti +  my_row.b_commission +
my_row. b_omission + my_row. b_theme_classification +
my_row.b_temporal_validity+my_row.b_topological_consi ;
 RETURN NEXT my_row;
END LOOP;
 RETURN;
END
$BODY$
LANGUAGE 'plpgsql';
//-------------------------------------------------------------------------
//Calculate the distance of dataset quality from users quality requirement
//-- Algorithm 17
CREATE OR REPLACE FUNCTION  ds_eaq_distance(INTEGER,INTEGER,CHARACTER)
RETURNS SETOF  tbltemp_eaq AS
$BODY$
DECLARE
r d_q__sub_element%rowtype;
```

```
my_row tbltemp_eaq%rowtype;
ur d_q__sub_element%rowtype;
userid alias FOR $1;
uext ALIAS FOR $2;
appn ALIAS FOR $3;
BEGIN
SELECT INTO r * FROM d_q__sub_element
WHERE user_profile_userprofile_id = userid  ;
FOR my_row IN
SELECT * FROM weighted_ds_quality_based_on_user_weight(userid,uext,appn)
LOOP
 my_row.d_absolute =
 r.d_q__abso_exter_posi_accur  -  my_row. d_q__abso_exter_posi_accur;
 my_row.d_omission =
 r.d_q__comp_omission - my_row.d_q__comp_omission;
 my_row.d_comission =
 r.d_q__com_commission - my_row.d_q__com_commission ;
 my_row.d_theme_classif =
 r.d_q__them_cassif_correct -my_row.d_q__them_cassif_correct ;
 my_row.d_tempo_varlidity  =
 r.dq__temp_validity-my_row.dq__temp_validity;
 my_row.d_topological_consi =
 r.d_q__logic_topo_consis - my_row.d_q__logic_topo_consis;
 my_row.area_ratio_per_user = ROUND( my_row.percentageofintrse);
 my_row.area_ratio_per_ds = ROUND(my_row.percentageofinterseperds);
 RETURN NEXT my_row;
END LOOP;
RETURN;
END
$BODY$
LANGUAGE 'plpgsql';
//-------------------------------------------------------------------------------
```

Appendix C

Spatial data recommendation

In this section the complete spatial data retrieval by calling the fitness for use evaluation functions explained in appendix B is given. This program gives spatial data recommendation based on user spatial data search quality input. The program accept user specific requirements and update profiles and retrieve datasets based on user input.

C.1 APPLICATION BASED SPATIAL DATA RECOMMENDATION

```
//=============================================================================
<?php
$user_id =  $_SESSION['u_id'] ; // Identify user in a session
$btnsearch = $_POST['btnsearch']; // accept new search button event
if ($btnsearch)
{
//create connection to the system database
$db_conn = pg_connect("host=itclx01 port=5432
dbname=sd_search user=tigist password=gi25461p");
if (!$db_conn) {
  echo "Failed connecting to postgres database $database\n";
  exit;
}
$myvar1 = $_POST['extent']; // accept user spatial extent requirement
// Identify index of extent table
$eqid = pg_query($db_conn,"select ex_boundingpolygon_id
FROM e_x__bounding_polygon " );
   $eqid_last = MAX( pg_fetch_all_columns($eqid));
   $eqids = $eqid_last + 1;
$extent = " INSERT INTO e_x__bounding_polygon
 (ex_boundingpolygon_id,the_geom,userprofile_id)
  VALUES ($eqids, GeomFromText('$myvar1',900913),$user_id)";
  $extentchk = pg_query($db_conn, $extent);

 // confirm  extent requirement
if(!$extentchk )
{
die("Error: No extent updated or specified").pg_last_error();
}
}
$planning ="planning"; // declaration of application name
// accept user application requirement
```

```
$applicationName = $_POST['applicationName'];
// check user application similar
similar_text($_POST['applicationName'],$planning,$percent);
if($percent != 100 )
{
$remember = $_SESSION['table_contents'];//  remember value for ranking
$items = array();
$sortcriteria = array();
if ($rank == "extent") {// check ranking  selection
$key = 0;
foreach ($remember as $data) {
$sortitem = sprintf('%020.15f %03d %03d', $data->area_ratio_per_user,
$data->relevance, $data->datasetname, $key);
$items[$sortitem] = array($data->sdr_uri, $data->relevance,
$data->area_ratio_per_user, $data->datasetname);
$sortcriteria[] = $sortitem;
++$key;
}
} else if ($rank == "application") {// check ranking selection
$key = 0;
foreach ($remember as $data) {
$sortitem = sprintf('%03d %020.15f %03d', $data->relevance,
 $data->area_ratio_per_user, $data->datasetname, $key);
$items[$sortitem] = array($data->sdr_uri, $data->relevance,
$data->area_ratio_per_user, $data->datasetname);
$sortcriteria[] = $sortitem;
++$key;
        }
}
sort($sortcriteria);
$sortcriteria=array_reverse($sortcriteria);
echo "<table border='1'><tr> <td>Recommended datasets</td>
 <td>Interseaction area ratio</td>
  <td>Application relevance</td></tr><tr></tr><br> ";
foreach ($sortcriteria as $sortitem) {
$data = $items[$sortitem];
echo "<tr><td>".'<a href="'.$data[0].'"  target="_blank"
      onmouseover="submitQuery('.$data[0].');">'.$data[3]."</a></td>
<td>".$data[2]."</td><td>".$data[1]." </td>";
}
echo "</table>";
}
else
{
$uext = pg_query($db_conn,
"SELECT ex_boundingpolygon_id From   e_x__bounding_polygon
WHERE userprofile_id =  $user_id AND ex_boundingpolygon_id= $eqids" );
$uextval = pg_fetch_array($uext);// fetch all the query
```

```
$uextvalues =$uextval[0] ;//Assign extent
$qu = pg_query($db_conn,
"SELECT sdr_uri,relevance,datasetname, area_ratio_per_user, the_deaom1
FROM extent_appliction_rec0( '$uextvalues','$applicationName')
order by w_application desc, percentageofintrse desc; ");
$chk = pg_num_rows($qu);
if($chk == 0){
        echo " No spatial datasets for the selected spatial extent" ;
}
else{
echo "<table border='0'>
 <tr>
 <td>Recommended datasets name </td> <td>Intersection area ratio</td>
 <td>Application relevance</td>
 </tr>";
 // initialize empty array to hold the recommended datasets
 $remember = array();
while ($data = pg_fetch_object($qu))
{
$remember [] = $data;
$x = $data->sdr_uri;
$y= $data->the_deaom1;
//Display the the recommendation  to user
echo "<tr><td>" . '<a href= " '.$x.'  onmouseover="deserialize();
" target="_blank" >'. $data->datasetname.'</a>',"</td><td>".$data->
area_ratio_per_user, "</td><td>".$data->relevance." </td></tr>";
}// assign the dataset to the session variable
$_SESSION['table_contents']=$remember;
pg_free_result($qu);// diplay result in free form
}
 echo "</table>";
}
// define a form for the ranking
$selectedfirst = "<option>extent</option>";
$selectedsecond = "<option>application</option>";
if ($rank == "extent")
{
$selectedfirst = "<option selected>extent</option>";
}
if ($rank == "application") {
$selectedsecond = "<option selected>application</option>";
}
?>
//=============================================================================
```

C.2 QUALITY BASED SPATIAL DATA RECOMMENDATION

The program given in this section allows users to search spatial data based on user specific quality requirement. When users specify quality requirement and weight the system use the user requirement to generate recommendation.

```
<?php
//------------------Check user quality requirements-------------
$user_id =  $_SESSION['u_id'] ;
$btnsearch = $_POST['btnsearch'];//accept new search button event
if ($btnsearch)
{
 $db_conn = pg_connect("host=itclx01 port=5432
 dbname=sd_search user=tigist password=gi25461p");
 if (!$db_conn) {
 echo "Failed connecting to postgres database $database\n";
   exit;
 }
 //Check user quality requirements
$allowedFields = array(
      'extent','applicationName', 'pext',
      'COm','cCom','tcl','tmpv','ltocon',
      'wpext','wcom','wccom', 'wtcl',
      'wtmpv','wltocon');
$requiredFields = array(
      'extent',  'applicationName','pext',
      'COm','cCom','tcl','tmpv','ltocon',
      'wpext','wcom','wccom','wtcl','wtmpv',
      'wltocon');
foreach($_POST AS $key => $value)
{
    // first need to make sure this is an allowed field
    if(in_array($key, $allowedFields))
    {
        $$key = $value;
      // is this a required field?
        if(in_array($key, $requiredFields) && $value == '')
        {
           $errors[] = "The field $key is required.";
        }
   }
}
if(count($errors) > 0)

{
    $errorString .= '<ul>';
    foreach($errors as $error)
  {
```

```
        $errorString .= "<li>$error</li>";
    }
    $errorString .= '</ul>';
header("location: searchbyquality_23.php");

}
}
?>
//------------------------Search by quality requirements-------------------------
<?php
$user_id =  $_SESSION['u_id'] ;
$myvar1 = $_POST['extent'];  // accept user extent
//echo "geom val is " .$myvar1;
// accept user quality requirement
$pext=$_POST['pext'];
$cOm =$_POST['cOm'];
$cCom =$_POST['cCom'];
$tcl = $_POST['tcl'];
$tmpv = $_POST['tmpv'];
$ltocon = $_POST['ltocon'];
// accept user quality requirement weight
$wpext=$_POST['wpext'];
$wcom =$_POST['wcom'];
$wccom =$_POST['wccom'];
$wtcl = $_POST['wtcl'];
$wtmpv = $_POST['wtmpv'];
$wltocon = $_POST['wltocon'];
//-----------------------------------------
$eqid = pg_query($db_conn,"SELECT ex_boundingpolygon_id
        FROM e_x__bounding_polygon " );
$eqid_last = MAX( pg_fetch_all_columns($eqid));
$eqids = $eqid_last + 1;
//-----------------------------------------
$extent = "INSERT INTO e_x__bounding_polygon
 (ex_boundingpolygon_id,the_geom,userprofile_id)
  VALUES ($eqids, GeomFromText('$myvar1',900913),$user_id)";
  $extentchk = pg_query($db_conn, $extent);
if(!$extentchk ) // confirm  quality requirement insertion
{
        die("Error: No extent stored").pg_last_error();
}
//-------------------------------------------------
$uext = pg_query($db_conn,
"SELECT ex_boundingpolygon_id
From e_x__bounding_polygon
WHERE
 userprofile_id =  $user_id AND ex_boundingpolygon_id= $eqids" );
$uextval = pg_fetch_array($uext);
```

```
$uextvalues = $uextval[0] ;
//-------------------------------------------------------
// Identify index of data quality element
//weight in the quality weight table
$wqid = pg_query($db_conn,"SELECT wt_dq_subelement_id
        FROM wt__d_q__sub_element " );
$wqid_last = MAX( pg_fetch_all_columns($wqid));
$wqids = $wqid_last + 1;
//-------------------------------------------------------
//Insert weight requirement into table
$wquality = " INSERT INTO wt__d_q__sub_element
(wt_dq_subelement_id, wt_d_q__abso_exter_posi_accurme,
 wt_d_q__comp_omission, wt_d_q__com_commission,
 wt_d_q__them_cassif_correct,  wt_d_q__temp_validity,
  wt_d_q__logic_topo_consis)
 VALUES ($wqids,$wpext,$wcom, $wccom, $wtcl, $wtmpv ,$wltocon)";
$wqcheck = pg_query($db_conn, $wquality);
if(!$wqcheck ) // confirm  quality requirement insertion
{
die("Error: No  quality weight is profiled").pg_last_error();
}
//Identify key of data quality elements
in the quality table for correct user
$qid = pg_query($db_conn,"SELECT dq_subelement_id
  FROM d_q__sub_element
  WHERE user_profile_userprofile_id= $user_id " );
 $qids =
  pg_fetch_all_columns($qid, pg_field_num($qid, 'dq_subelement_id'));
 $qidrow = pg_num_rows($qid);
 $quser = $qids[0];
 if($qidrow !=0){
 $qryupdate = "UPDATE d_q__sub_element
 SET  d_q__abso_exter_posi_accur= '".$pext."' ,
 d_q__comp_omission='".$cOm."',
 d_q__com_commission='".$cCom."',
 d_q__them_cassif_correct='".$tcl."',
 dq__temp_validity='".$tmpv."',
 d_q__logic_topo_consis ='".$ltocon."',
 user_profile_userprofile_id= '".$user_id ."',
 wt__d_q__sub_element_wt_dq_subelement_id='".$wqids."'
 WHERE dq_subelement_id='".$quser."'";
 $upd = pg_query($db_conn,$qryupdate);
if(!$upd )
{
die("Error: No update performed").pg_last_error();
}
}
else {
```

```
//Insert quality requirement into table
$qryqid = pg_query($db_conn,
"SELECT dq_subelement_id from d_q__sub_element " );
$qid_last = MAX( pg_fetch_all_columns($qryqid));
$qidn = $qid_last + 1;
$quality = "INSERT INTO d_q__sub_element (dq_subelement_id,
d_q__abso_exter_posi_accur, d_q__comp_omission,
d_q__com_commission,d_q__them_cassif_correct,
dq__temp_validity, d_q__logic_topo_consis,
user_profile_userprofile_id,
wt__d_q__sub_element_wt_dq_subelement_id)
VALUES
($qidn, $pext,$cOm,$cCom,$tcl, $tmpv, $ltocon, $user_id ,$wqids)";

$qcheck = pg_query($db_conn, $quality);

if(!$qcheck ) // confirm  quality requirement insertion
{
die("Error: No user quality is inserted").pg_last_error();
}
}
$qu = pg_query($db_conn, "SELECT *
FROM ds_quality_distance(".$user_id.",".$uextvalues.")
ORDER BY percentageofintrse desc; ");
echo "<br />";
echo "<table border='1'>
<br />
 <tr><td>
Recommended datasets name </td><td>Intersection </td>
<td> weight aggregated </td>
 </tr>";
while ($data = pg_fetch_object($qu)) {
        echo "<tr>";
        $x = $data->sdr_uri;
echo "<td>".'<a href= " '.$x.'" target="_blank">'.
         $data ->datasetname.'</a>'.
" </td><td>". $data -> area_ratio_per_user."</td>
<td>". $data -> weight_agg."</td>";
}
pg_free_result($qu);
echo "</tr></table>";
?>
//=============================================================================
```

C.3 APPLICATION AND QUALITY BASED SPATIAL DATA RECOMMENDATION

This section of the program used by the system to retrieve spatial data based on user application requirement and quality requirements.

```
//=============================================================================
<?php
$myvar1 = $_POST['extent'];  // accept user extent
//accept user quality requirement
$pext=$_POST['pext'];
$cOm =$_POST['cOm'];
$cCom =$_POST['cCom'];
$tcl = $_POST['tcl'];
$tmpv = $_POST['tmpv'];
$ltocon = $_POST['ltocon'];
$applicationName = $_POST['applicationName'];
//accept user quality requirements weight
$wpext=$_POST['wpext'];
$wcom =$_POST['wcom'];
$wccom =$_POST['wccom'];
$wtcl = $_POST['wtcl'];
$wtmpv = $_POST['wtmpv'];
$wltocon = $_POST['wltocon'];
//Identify index and set it on new row
$eqid = pg_query($db_conn,"select ex_boundingpolygon_id
 FROM e_x__bounding_polygon " );
$eqid_last = MAX( pg_fetch_all_columns($eqid));
$eqids = $eqid_last + 1;

//Identify index of data quality element weight from table
$wqid = pg_query($db_conn,"select wt_dq_subelement_id
FROM wt__d_q__sub_element " );
$wqid_last = MAX( pg_fetch_all_columns($wqid));
$wqids = $wqid_last + 1;

//Identify the maximum index in the quality table
$qryqid =
pg_query($db_conn,"select dq_subelement_id from d_q__sub_element " );
$qryqid_last = MAX( pg_fetch_all_columns($qryqid));
$qids = $qryqid_last + 1;

//system learn user extent requirement from interface
$extent = "INSERT INTO e_x__bounding_polygon(ex_boundingpolygon_id,
the_geom,userprofile_id)
VALUES ($eqids, GeomFromText('$myvar1',900913),$user_id)";
$extentchk = pg_query($db_conn, $extent);

// confirm  spatial data search quality requirements profiling
if(!$extentchk )
```

```
{
        die("Error: No extent stored").pg_last_error();
}
$uext = pg_query($db_conn,
"SELECT ex_boundingpolygon_id From   e_x__bounding_polygon
WHERE userprofile_id =  $user_id AND ex_boundingpolygon_id= $eqids");
$uextval = pg_fetch_array($uext);
$uextvalues = $uextval[0] ;

//Insert weight requirement into table
$wquality = " INSERT INTO wt__d_q__sub_element
(wt_dq_subelement_id, wt_d_q__abso_exter_posi_accurme,
wt_d_q__comp_omission, wt_d_q__com_commission,wt_d_q__them_cassif_correct,
wt_d_q__temp_validity, wt_d_q__logic_topo_consis)
VALUES ($wqids,$wpext,$wcom, $wccom, $wtcl, $wtmpv ,$wltocon)";
   $wqcheck = pg_query($db_conn, $wquality);
// confirm  quality requirement insertion
if(!$wqcheck )
{
        die("Error: No weight quality is stored").pg_last_error();
}
//Search quality requirement from user profile
$qid = pg_query($db_conn," select dq_subelement_id from d_q__sub_element
WHERE user_profile_userprofile_id= $user_id " );
//fetch the query
$qids = pg_fetch_all_columns($qid, pg_field_num
($qid, 'dq_subelement_id'));
$qidrow = pg_num_rows($qid);
$quser = $qids[0];
//Check user quality requirement found
IF($qidrow !=0)
{
//update user quality requirement
$qryupdate =
 "UPDATE d_q__sub_element SET
  d_q__abso_exter_posi_accur= '".$pext."' , d_q__comp_omission='".$cOm."',
  d_q__com_commission='".$cCom."',d_q__them_cassif_correct='".$tcl."',
  dq__temp_validity='".$tmpv."', d_q__logic_topo_consis ='".$ltocon."',
  user_profile_userprofile_id= '".$user_id ."',
  wt__d_q__sub_element_wt_dq_subelement_id='".$wqids."'
  WHERE dq_subelement_id='".$quser."'";
        $upd = pg_query($db_conn,$qryupdate);

// confirm  quality requirement update
IF(!$upd )
{
die("Error: Quality requirement is not updated").pg_last_error();
}
```

```
}
ELSE {
//Insert quality requirement into table
$quality = " INSERT INTO d_q__sub_element
(dq_subelement_id, d_q__abso_exter_posi_accur,
d_q__comp_omission, d_q__com_commission,d_q__them_cassif_correct,
dq__temp_validity, d_q__logic_topo_consis, user_profile_userprofile_id,
wt__d_q__sub_element_wt_dq_subelement_id)
VALUES ($qids, $pext,$cOm,$cCom,$tcl, $tmpv, $ltocon, $user_id ,$wqids)";
$qcheck = pg_query($db_conn, $quality);
// confirm  quality requirement insertion
if(!$qcheck )
{
die("Error: No user quality is stored").pg_last_error();
}
}
$planning ="planning";
similar_text($_POST['applicationName'],$planning,$percent);
if($percent != 100 ){
echo "<br /> <br /> No dataset recommendation for this application";
}
// call quality evaluation function
$qu = pg_query($db_conn,
 "SELECT * from ds_eaq_distance
       (".$user_id.",".$uextvalues.",'".$applicationName."')
  ORDER BY area_ratio_per_user desc, weight_agg desc");
     $array = pg_fetch_all_columns($qu, pg_field_num($qu, 'sdr_uri'));
echo "<table border='1'> <tr>
 <td>Recommended datasets</td><td> Area ratio inter/user </td>
 <td> Area ratio inter/ds </td>
 <td> aggregated quality</td><td> Application relevance</td>
 </tr>";
 //fetch the returned query as object
while ($data = pg_fetch_object($qu)) {
echo "<tr>";
$x = $data->sdr_uri;
echo "<td>" . '<a href= " '.$x.'" target="_blank">'. $data ->
datasetname.'</a>'. " </td> <td>".$data->area_ratio_per_user, "</td>
<td>".$data-> area_ratio_per_ds, "</td><td>".$data->
weight_agg , "</td><td>".$data-> relevance, "</td>";
}
pg_free_result($qu);
echo "</tr></table>";
?>
//=========================================================================
```