COMBINED MAP PERSONALISATION ALGORITHM FOR DELIVERING PREFERRED SPATIAL FEATURES IN A MAP TO EVERYDAY MOBILE DEVICE USERS

Avinash Turab Bookwala

A thesis submitted to

Auckland University of Technology

in fulfilment of the requirements for the degree of

Master of Engineering (ME)

May, 2010

School of Engineering

DEDICATION

To my wonderful, beautiful and most beloved wife,

Sandini

To my caring and supporting parents,

Bachoo and Turab

and

To my most considerate and helpful brother,

Nakib

ACKNOWLEDGEMENTS

Firstly, I would like to thank God for all His abundant blessings, guidance and assistance throughout my life and especially through this thesis, without which I would never have got to where I am today.

My utmost gratitude is owed to my supervisor, Dr. Helen Zhou, for enabling me to work with her in this very new and exciting area of research. I thank her for her assistance, wisdom, and support, without which the goals set out for this project would not have been achievable. I also thank her for constantly pushing me towards working harder and exercising an eye for quality. I am grateful for all her time and energy dedicated towards the improvement of this thesis and this research as a whole.

I would also like to thank Prof. Ahmed Al-Jumaily and Dr. John Collins for their time and effort spent towards the completion of this thesis. Their knowledgeable assistance has been very useful and is greatly appreciated. Thanks to the administrative staff at IBTec (Institute of BioMedical Technologies) for their support and guidance in various aspects of this master's degree. I thank Mr. Justin Zheng and Mr. Wenjie Li, the industrial mentors of this thesis, at Goldfinger HighTech Ltd., Auckland, for providing me with the map data required for the execution of this project. I would also like to them for their friendly, supportive nature towards me at all times.

The financial support offered by Foundation for Research, Science and Technology (FRST) through the Technology for Industry Fellowship (TIF) grant, is acknowledged. To all the users who tested and evaluated the system, your co-operation and help offered in this regard is highly appreciated.

I would also like to acknowledge the support and incessant love given to me by my life partner, Sandini, and for constantly listening and trying to understand my various discussions about this thesis. I would like to thank her for always being there for me, believing in me and giving me the confidence needed to complete this thesis. Thanks to my parents, Bachoo Bookwala and Turab Bookwala, for their moral support and aid in every way possible, and also for their efforts in proofreading this thesis. I deeply appreciate the constant support and encouragement from my brother, Nakib Bookwala, and would like to thank my dear friends, Curwin Lee Boltman and Sanduni Dheerasooriya, for their compassion and assistance throughout the phase of this master's research.

ABSTRACT

In this thesis, we present an innovative and novel approach to personalise maps/geo-spatial services for mobile users. With the proposed map personalisation approach, only relevant data will be extracted from detailed maps/geo-spatial services on the fly, based on a user's current location, preferences and requirements. This would result in dramatic improvements in the legibility of maps on mobile device screens, as well as significant reductions in the amount of data being transmitted; which, in turn, would reduce the download time and cost of transferring the required geo-spatial data across mobile networks. Furthermore, the proposed map personalisation approach has been implemented into a working system, based on a four-tier client server architecture, wherein fully detailed maps/services are stored on the server, and upon a user's request personalised maps/services, extracted from the fully detailed maps/services based on the user's current location, preferences, are sent to the user's mobile device through mobile networks. By using open and standard system development tools, our system is open to everyday mobile devices rather than smart phones and Personal Digital Assistants (PDA) only, as is prevalent in most current map personalisation systems.

The proposed map personalisation approach combines content-based information filtering and collaborative information filtering techniques into an algorithmic solution, wherein content-based information filtering is used for regular users having a user profile stored on the system, and collaborative information filtering is used for new/occasional users having no user profile stored on the system. Maps/geo-spatial services are personalised for regular users by analysing the user's spatial feature preferences automatically collected and stored in their user profile from previous usages, whereas, map personalisation for new/occasional users is achieved through analysing the spatial feature preferences of like-minded users in the system in order to make an inference for the target user. Furthermore, with the use of association rule mining, an advanced inference technique, the spatial features retrieved for new/occasional users through collaborative filtering can be attained. The selection of

spatial features through association rule mining is achieved by finding interesting and similar patterns in the spatial features most commonly retrieved by different user groups, based on their past transactions or usage sessions with the system.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

GIS	**G**eographical **I**nformation **S**ystem
GPS	**G**lobal **P**ositioning **S**ystem
GUI	**G**raphical **U**ser **I**nterface
J2ME	**J**ava **2** Platform, **M**icro **E**dition
JDBC	**J**ava **D**ata**B**ase **C**onnectivity
JSP	**J**ava**S**erver **P**ages
NDM	**N**etwork **D**ata **M**odel
Oracle AS	**Oracle A**pplication **S**erver
PDA	**P**ersonal **D**igital **A**ssistant
URL	**U**niform **R**esource **L**ocator
WAP	**W**ireless **A**pplication **P**rotocol
XML	**E**xtensible **M**arkup **L**anguage

STATEMENT OF ORIGINALITY

‘I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person, nor material which to a substantial extent has been accepted for the qualification of any other degree or diploma of a university or other institution of higher learning, except where due acknowledgment is made in the acknowledgments.’

....(signed)

....(date)

CHAPTER - 1

INTRODUCTION

1.1 MOTIVATION

A Geographic Information System (GIS) is a special type of computer-based information system tailored to store, process and manipulate geo-spatial or geographically referenced data [57]. It constitutes of an integrated tool box for spatial data input, storage, management, retrieval, manipulation, analysis, modelling, output, and display [34]. GIS technology has various uses, such as resource management, environmental impact assessment, urban planning, criminology and many more [2, 12, 33, 52]. All applications of GIS technology typically consist of displaying maps/geo-spatial services to their users. Even in our daily life, maps/geo-spatial services have been widely used from personal vehicle navigation, business analysis, dog tracking systems, to fitness and hiking devices.

Traditionally, users used their desktop Personal Computers (PCs) to access maps/geo-spatial services through a GIS via the Internet. With the recent development of high speed mobile networks such as 3G/3.5G, users expect to access these maps/services using their everyday mobile devices (i.e., Personal Digital Assistants (PDAs), smart phones, average mobile phones), via ubiquitous mobile networks, anywhere and anytime [88].

Due to the large size of geo-spatial data, currently a number of commercially available mobile GIS[1] applications or services are mainly installed on particular smart phones and PDAs [81] with larger screens, greater memory capacities, and a wider range of input methods as compared to average everyday mobile devices. Selection of relevant spatial feature content (motorways, petrol stations, shopping malls etc.) in maps, based on a user's preferences and requirements, called personalisation, has been acknowledged as a viable approach to providing everyday mobile devices convenient access to a mobile GIS [1, 22, 35, 36, 43, 44, 45]. Using this map personalisation approach, spatial features relevant to the user, based on their preferences, are extracted from a detailed geo-spatial dataset, and delivered as a personalised map to the user's mobile device. Therefore, through the personalisation of maps, the amount of spatial information displayed on mobile devices' small screens is significantly reduced, making the displayed maps more legible, and reducing the information overload. This, in turn, would also reduce the time taken and high cost to transmit such data across costly mobile networks and the Internet. The main focus of this thesis is based on the delivery of personalised maps to everyday mobile device users, taking into account their current location, preferences and requirements, as this would encourage the use of mobile GIS applications in our daily life.

1.2 PROBLEM DEFINITION

A few commercially available GIS come pre-installed in proprietary handheld devices, normally requiring fast processors and large memory capabilities. With pre-loaded maps/geo-spatial services, a proprietary handheld GIS can work by itself, without connection to a costly wireless network. However those GIS are typically designed for specific application purposes such as in-car navigation systems, asset tracking systems and field data loggers. A description of a typical handheld GIS,

[1] Mobile GIS is a Geographical Information System based on mobile computing and mobile Internet. It is designed as an extension of the web GIS to mobile Internet, including wireless internet and mobile communication networks [79].

discussing its useful features, can be found in Appendix B. Also, they suffer from delay on updating the datasets/systems pre-installed on the handheld devices.

Recently, with the development of 3G/3.5G mobile networks, mobile GIS have been developed based on distributed client server system architectures, wherein comprehensive maps/geo-spatial services are stored and maintained in a server, and upon a user's request, can be downloaded from the server and sent to the user across the Internet and mobile networks [1]. These client-server mobile GIS have addressed the issues related to proprietary handheld GIS such as:

- Only proprietary handheld devices being able to access their applications
- Designated application purposes, and
- Delay on updates for pre-installed datasets/systems

However, such mobile GIS are available to specific smart phones and PDAs only and still suffer from slow transmission and high mobile network costs, due to large geo-spatial data transmitted from the server to clients. On the other hand, maps downloaded on the client side often contain a lot more information than what users actually need (called information overload). Since everyday mobile devices possess limited capabilities, such as small display screen sizes, limited memory capacities, and limited application development technology compared to smart phones and PDAs [79], the issue of information overload proves to be quite significant. Therefore, mobile GIS applications/services for use with everyday mobile devices need to take into account their limitations when being developed.

Map personalisation has recently been proposed [5, 22, 35, 44, 75, 88] and investigated to address these issues, whereby geo-spatial features are to be tailored for individual users. Using map personalisation, only relevant geo-spatial features will be extracted from comprehensive geo-spatial datasets, and delivered to the user as a personalised map. Some of the advantages of incorporating such map personalisation in a mobile GIS include, i) improved legibility of maps on everyday mobile devices' small screens, and ii) reduction in the amount of data sent across mobile networks, offering faster and cheaper map downloads. There has been a considerable amount of research conducted in this field; however, map personalisation is still an open research topic due to lack of formalisation in developing user preference acquisition techniques.

Two information filtering techniques, content-based filtering and collaborative filtering, have been proposed, in which content-based filtering has been used to personalise maps based on an individual user's preferences stored in their user profile. Whereas, collaborative filtering techniques have been used to retrieve spatial feature content for a target user, based on the preferences of other like-minded users. Like-minded users in this thesis are defined as a group of users sharing similar interest in particular spatial features, and can be determined by analysing the spatial features retrieved by users in each session of usage. However, both techniques have their disadvantages; it is not feasible to implement content-based filtering to personalise maps for new or occasional users without having any record of their user profile, and collaborative filtering is unable to support the detailed information filtering available through content-based filtering. Therefore, a combined approach to map personalisation, accommodating for regular users with detailed user profiles, through content-based filtering, and new/occasional users without user profiles, through collaborative filtering, would prove to be a significant contribution to this field of research.

1.3 RESEARCH OBJECTIVES

This research aims to address the issue of excessive amounts of irrelevant geo-spatial data being transferred to users of a mobile GIS, by incorporating map personalisation, which extracts only information relevant to the user from a fully detailed map dataset, based on the user's location, preferences and requirements. This process of map personalisation provides a significant reduction in the amount of geo-spatial data transmitted, resulting in reduced time and cost of geo-spatial data transmission, as well as improved map legibility on mobile devices' small screens. Moreover, with the use of standard system development tools, the proposed GIS is able to support a wide range of mobile devices, including everyday mobile phones as well.

The following provides a summary of the main objectives of this thesis:

- To develop a combined map personalisation algorithm providing personalised maps to users based on their spatial feature preferences, by utilising content-based filtering for existing users and collaborative filtering for new/occasional users
- To incorporate association rule data mining, an advanced inference technique, for collaborative information filtering
- Integrate the combined map personalisation algorithm into a working system, open to a wide variety of everyday mobile devices
- Design an easy to use Graphical User Interface (GUI) for mobile users, regardless of their skills and age

1.4 RESEARCH METHODOLOGY AND THESIS CONTRIBUTION

Combined Map Personalisation Approach

User profiles can be acquired explicitly and implicitly. For instance, a questionnaire has been acknowledged as a conventional explicit way to obtain initial user profiles. However, very few people are keen to complete a time-consuming questionnaire. Moreover, a user's preferences are likely to change over time [42], making it almost impossible to keep track of such changes by continually asking the user to provide filled-in forms. Automatically or implicitly recording a user's preferences in a "web cookie" (a small unique text string), has been widely used by web servers to differentiate users and maintain specific information about these users (e.g. site preferences and shopping transactions). Cookies have the potential to be deliberately changed [20] and therefore, the information relating to users is not always reliable. Recently, conceptual models for web usage mining systems have been proposed to implicitly collect user movements and interactions on a website for further analysis of user profiles [49, 83, 84, 87]. However, analysing users' implicit interactions could lead to inaccurate assumptions of a user's preferences, due to the user not directly specifying their preferences.

Current personalisation approaches are mainly based on either content-based information filtering or collaborative information filtering techniques. For regular users with a user profile, content-based filtering approaches have been used to select relevant information appropriate to the user's profile [71]. However, it is not possible for content-based filtering approaches to provide relevant information to new or occasional users, without having any record of their preferences in a user profile. On the other hand, collaborative filtering methods utilise the preferences of like-minded users, in making a prediction for a target user [11, 41]. Relevant information can then be passed between those like-minded users, at an acceptable level of accuracy. Although collaborative filtering improves the efficiency of filtering large amounts of information, it does not support detailed information filtering through contents, and is not feasible in cases of new systems, where no user profiles have yet been stored [62].

In order to address these issues with content-based or collaborative filtering techniques, we have proposed a map personalisation algorithm that combines content-based filtering for regular users having a user profile stored, and collaborative filtering for occasional users having no user profile stored. Such a combined approach will result in an effective and efficient algorithm [7, 62] for the personalisation of maps/geo-spatial services.

Association Rule Mining for Collaborative Filtering

Using collaborative filtering techniques for map personalisation enables the retrieval of spatial feature content for a target user, based on the preferences of like-minded users. Furthermore, using association rule mining, we can determine the probability that a target user would be interested in particular spatial features, based on the spatial features commonly retrieved together by other similar users. This probability can then be used to determine the relevance of particular spatial features for a target user.

Standard System Development Tools

The proposed combined map personalisation algorithm will be integrated into a working system. The system will be based on a distributed client-server architecture, including mobile users, a Wireless Application Protocol (WAP) gateway, an application server, and a map server. This system will be developed using open and standard system development tools, such as Java programming language, enabling it to be interoperable with other mobile GIS as well as allowing flexibility and easy adaptation into other environments.

Standard communication protocols, such as HyperText Transfer Protocol (HTTP) and WAP have been adopted for communicating between the mobile users and the map server, allowing cross-platform and hardware independent access across the Internet and mobile networks, anywhere and anytime.

Easy to Use Graphical User Interfaces

For high portability, average everyday mobile devices are typically equipped with small display screens, constrained input/output methods, and limited application development scope as compared to PDAs and smart phones. An easy to use GUI will be designed and developed for mobile devices, taking into consideration their limitations in screen size, input methods, and technology scope. A GUI for desktop PC users will be initially developed and intended for more complex GIS functions and for administrative use.

Thesis Contribution

The contribution of this thesis is mainly concentrated in the field of map personalisation for everyday mobile devices. We propose a novel approach to achieve the personalisation of maps, through combining two well known information filtering techniques of content-based filtering and collaborative filtering, along with the inclusion of an advanced inference technique of association rule mining to provide tailored maps/geo-spatial services to users. Our combined map personalisation algorithm has addressed some common issues related to the field of map personalisation, such as information overload on the small screens of mobile devices, and large amounts of spatial data being transferred across costly mobile networks. With the implementation of a working system based on a distributed client-server architecture, and the utilisation of open and standard system development tools, enables a wide variety of everyday mobile devices to access the system across mobile networks and the Internet, anytime and anywhere. All the objectives set up for our research have been successfully achieved.

1.5 SCOPE OF THIS THESIS

Given below is a brief outline of the chapters covered in this thesis. Chapter 2 presents some related research work conducted in the fields of user profile acquisition and map personalisation. It includes research conducted on the acquisition of user profiles through users' implicit and explicit interactions, as well as how these user profiles can be analysed and interpreted to personalise maps.

Our combined map personalisation algorithm will be detailed in Chapter 3. This chapter investigates into the combination of content-based and collaborative information filtering techniques into an algorithmic solution for map personalisation. Furthermore, it discusses in detail the theory of association rule mining, and its incorporation with collaborative filtering to personalise maps for occasional users with no user profile.

Chapter 4 discusses the architecture of the developed system, along with the tools used for implementation. The map datasets used and the various tables created in order to facilitate the correct functioning of the system, are also discussed. The implementation of the proposed combined algorithm for map personalisation into our system is discussed. This chapter also includes an explanation of the functionalities of the GUI developed for everyday mobile device users.

In Chapter 5, our system evaluation is carried out in terms of data reduction, processing time and map personalisation, revealing promising results. The everyday mobile devices used for the system evaluation are discussed. To test our combined map personalisation algorithm, the implemented system was evaluated by a group of selected users, regardless of their age and skills. The results of the map personalisation evaluation have also been presented in this chapter, along with a discussion on the interoperability of our system with other existing mobile GIS.

Chapter 6 provides a conclusion for this thesis. It outlines our achievements and discusses some potential further research work that could branch out from this thesis.

CHAPTER - 2

PREVIOUS AND RELATED WORK

2.1 INTRODUCTION

The personalisation of maps on mobile devices traditionally allowed a user to change the colour of the interface components, fonts [21] or edit the representation of features on the map, such as altering the colour, width and style of map features. Moreover, an additional modification could be setting up favourite locations specified by the user, such as the user's home address, favourite restaurant, business/work address, etc.

Map personalisation approaches have been recently proposed [5, 22, 35, 44, 75, 88] to extract relevant geo-spatial information based on a user's current location, preferences and requirements. By taking each individual user's current location, preferences and requirements into consideration, the personalised map/service is a much simplified representation extracted from a fully detailed map/geo-spatial service. This leads to a remarkable reduction in the amount of geo-spatial information sent to the user, thus preventing information overload (only relevant data is displayed on the map) and improving map legibility on mobile device screens. However, such map personalisation approaches have mainly been developed for selected PDAs and smart phones only. Average everyday mobile devices though have a few limitations, such as their small screen sizes, restricted memory capabilities, and limited technology scope.

Such map personalisation relies on the creation of user profiles. However, the formalisation of user profile acquisition is still lacking. Traditionally, user profiles were acquired based on explicit user interaction with an interface, wherein the user directly specifies his/her preferences regarding map feature content, through some form of questionnaire or survey. This method is often quite time consuming and users may not be interested in filling out a questionnaire or survey. More recently, methods of acquiring user profiles implicitly have been proposed, by monitoring and recording a user's implicit interaction with map features. This information can then be analysed to understand users' preferences regarding spatial content retrieval, and only retrieve spatial content of significant relevance to the user.

The remainder of this chapter is structured as follows. Section 2.2 discusses some previous and related work in the acquisition of user profiles. Section 2.3 provides a detailed literature review in personalisation, with a focus on map personalisation. Finally, in Section 2.4, our proposals to address some of the common issues in this field of research are outlined and discussed.

2.2 USER PROFILE ACQUISITION

Currently, user profiles are acquired based on collecting interactions between users and the features of a map interface, categorised into explicit and implicit interactions. User profiles acquired through users' explicit interactions with a map interface rely solely on the user directly specifying their current requirements or preferences. Collecting users' explicit interactions is quite often implemented in a manner where the user is requested to complete a questionnaire or survey about their preferences or requirements. However, very few people are keen to complete a time-consuming questionnaire. And also, "the collected preferences would be typically only a subset of the user's real needs and wants, because users usually do not like to input data, and GUI designers do not want to bother users with many questions." [65] Moreover, the user's preferences are likely to change over time [42], and it is almost impossible to keep track of such changes by asking the user to continually provide filled-in forms. Therefore, in order to overcome these issues of explicitly requesting information from the user, new techniques of implicitly collecting a user's interests or preferences have been developed, based on their multi-modal interactions (i.e., mouse clicks, pen/stylus based interaction, and speech instruction recognition) with map features [21, 35, 36]. However, the user's implicit interactions with map features via the pen/stylus or speech instructions is restricted for use with either Tablet PCs, or certain PDAs and smart phones that offer touch screen capabilities. An example of collecting a user's preferences through their implicit interactions could be, when a user disables a particular feature on the map, the assumption is made that the user is not interested in that feature; therefore, the next time the user logs in, that particular feature would not be given preference of loading on the map. By capturing a user's explicit and implicit interactions, a suitable model for personalisation can be developed to a satisfactory level of accuracy. Some past research conducted in this area is discussed below.

An example of recent work conducted using explicit user interactions to create and store user profiles is outlined in De Carolis et al. [5]. In [5], a system has been developed to provide maps to a user with personalised features or objects of interest. The application is developed for a PDA and uses a user model ontology developed by UbisWorld [78] to manage the mobile user profiles in the system. These user profiles are managed via a GUI, wherein the user is required to explicitly specify their preferences and interests. Based on some user defined attributes, such as time of the day (lunch, breakfast, or dinner time), current location, and purpose of usage (such as business or leisure), the system is able to make a recommendation of, for example, the type of restaurants (fast food, fine dining etc.) to display on the map, based on the user's current location and preferences available in their user profile.

The PDA application displays a map within the user's current location, containing the objects of interest or landmarks based on the user profile. The user profiles are stored on the PDA itself, along with the geo-spatial maps/services used in the system. However, storing the spatial data on the PDA makes it difficult for the user to update this data when new map data is made available. This is due to high costs of data updates, as well as the time delay in receiving the data updates from the suppliers.

Moreover, the system relies on the mobile user profile ontology and interface developed by UbisWorld, which requires a substantial amount of user input to build up the user profiles, with specification of their preferences and interests. Filling out questionnaires regarding user preferences and interests is often quite time consuming, and an unfavoured approach to retrieve such information. Also, it would need frequent updating as users' preferences may change from time to time, as discussed earlier.

In [16, 56], Rinner and Raubal investigate into the feasibility of evolving simple spatial and attribute queries available in typical Location-Based Services (LBS), into a multi-criteria evaluation method based on user preferences. In doing so, traditional LBS can be offered to users enhanced by user's specifying multi-criteria ordered weighted values for the services. This research, however, relies on explicitly acquiring user input regarding specifying ordered weight values for non-spatial attributes, used as evaluation criteria in the recommendation of spatial and non-spatial information.

The above discussed systems all mainly rely on explicit user input in order to achieve map personalisation. However, this method of collecting user preference information can be quite bothersome to the majority of users, especially users of average mobile devices with their limited capabilities, such as input restrictions and small display screens.

Tezuka and Tanaka [75] developed a system that monitors users' implicit interactions to personalise a map. The system estimates users' intentions regarding their navigation of the map, based on their operation history. The trajectory, which is a series of users' panning operations history, is analysed, and their intentions regarding future map panning operations are inferred. In this system, when a route is planned for a user from a starting point to a destination, it is assumed that the user may pan the map either along the route to the destination or straight across to the destination. In the first panning operation type the map tiles are constantly generated, whereas in the latter panning operation type the system waits till the user reaches the destination to send only that area of the map, reducing download cost and time. The system also displays landmarks that are within a certain range out of the map view area. These are displayed surrounding the edges of the map, in the direction of the landmark location relative to the position of the map. The enabling or disabling of surrounding landmarks is based on the user's intentions, established from their panning operation history.

The system mentioned above is able to estimate users' intentions regarding their map panning operations to determine whether they are interested in retrieving certain map frames or not. This method reduces the amount of map data sent across to the user's terminal, decreasing the time taken to transmit the spatial data and the cost of transmission.

User's panning intentions is one possible technique to personalise map spatial content retrieval. However, by developing the system to infer users' preferred spatial objects, themes, or amenities of interest, would enhance the personalisation of spatial content retrieval currently offered in the system. The time taken for map generation would, as a result decrease, due to only map features of relevance to the target user being rendered.

Monitoring users' implicit interactions with the mapping interface through their mouse movements has been proposed in [22, 23]. Mac Aoidh [23] developed a system which analyses users' mouse movement behaviour while using a geo-spatial mapping interface. Users' mouse movements and operations are examined to implicitly determine their interests, and personalise spatial content for them accordingly. The parameters of a mouse interaction that are collected and analysed by the system include; the latitude and longitude of the mouse cursor over the map, its duration in each position, the co-ordinates and contents of each map view, and its scale. Further information, such as the time taken or time difference in performing panning and zooming actions, is also recorded and analysed to determine user behaviour while browsing maps. The user's mouse interaction data is then fed into a simple algorithm developed in [45], to output a list of spatial objects (points, polylines, and polygons) on the map that are most likely to be of interest to the user. The objects deemed of interest can then be visualised as an overlaid layer placed on top of the currently displayed map.

This system is able to produce interactive visual descriptions, revealing hidden and possibly valuable information about system users, which can be used by system designers and providers to gain a better understanding of user preferences regarding particular map objects. Obtaining this information implicitly, without interrupting the user, is quite useful, as users most often do not want to be bothered with explicitly providing information about their preferences. However, analyzing a user's mouse movements and interactions with the interface has the possibility of untrue information about a user being collected, since a user may unintentionally leave the mouse pointer at places of little or no interest. An advanced reinforcement method needs to be implemented in order to improve the inference of objects deemed of interest for the user. Personalisation is currently offered at a very broad level, personalising maps based on spatial objects (points, polylines, and polygons). More detailed level of personalisation, such as personalisation of map spatial feature content (highways, city streets, police stations etc.) would be beneficial in understanding a user's preferences or interests.

Combining the techniques of capturing implicit and explicit user interactions, prior to storing these interactions into user profiles, provides an improved understanding of a user's preferences regarding spatial information retrieval [47]. The following work discusses a framework for the combined user profile acquisition approach.

In [87] Jrad et al. discusses the architecture of the Eiffel Project, which focuses on the modelling of user profiles and their contexts. The user profile models created in this project collect information through the implicit and explicit interactions of users. Each user profile model consists of the user's preferences and some personal details, such as computer name, time, place, history, etc. This system aims to capture the users' explicit as well as implicit interactions. Explicit interactions are captured in static user profiles through either, a questionnaire requesting the user's contextual information, the user's system registration, or the user's feedback regarding a recommended activity to determine the preference of that particular activity. Implicit interactions are captured in dynamic user profiles through; i) cookies stored on the client computer, ii) logs containing information such as client computer name, date/time and resource accessed, and iii) through feedback regarding the client's interaction with the information automatically recommended to him/her. The system groups together users having similar interests, and constantly monitors the user's implicit and explicit interactions storing them in their respective profiles.

If the user queries an object, the system looks through the user profile to find any other information that might be relevant to that query and offers it to the user, for e.g., if the user is looking for hotels and somewhere in his profile it is stated that he prefers air-conditioned rooms, then the system would automatically add that condition to the query. The system also recommends places of interest, and offers information about the location that the user is in, based on the information stored in the user profiles. The algorithm used for the inference and recommendation of items is able to, i) track user behaviour to recommend similar items based on previous item choices, ii) find similarities in different users' preferences and group them together, iii) implement rules such as association rules to recommend similar items, by determining a user's probability of being interested in a particular landmark, based on their previous visits to similar landmarks, and iv) utilise web usage data mining techniques to extract and interpret the data implicitly stored in the user log files.

Although the architecture for the Eiffel Project has been clearly detailed, the interface has yet not been developed and only claims have been made about the layout of the interface. The system is being developed for PC users only and is implemented only for web content personalisation, rather than spatial content.

Another system known as CRUMPET has been designed and validated in [10] focusing on offering personalisation of location-aware tourist services, based on a multi-agent system. CRUMPET is a mobile GIS that offers recommendations of services such as tourist attractions to individual users, based on their personal interests and current location via GPS (Global Positioning System). User profiles, in this system, are collected either, through users' implicit interactions with the spatial objects, or through them explicitly specifying their interest in particular spatial objects. However, it does not accommodate for new/occasional users having no user profile stored, which is one of the objectives of our thesis. Also this system is limited for use with a PDA, due to their larger screen size and technical advantage over average mobile devices.

More recently, a working system has been developed by Doyle et al. [35, 36] to explicitly and implicitly collect user interactions with map features by utilising stylus based interactions and speech instructions recognition. With the use of latest user interaction techniques supported by Tablet PCs and advanced PDAs, the interactions between users and map features can be dramatically enhanced by a single stylus click or verbal command. All the user interactions are recorded at a remote server. An individual user profile can be created and updated accordingly. Based on the user profile, only relevant map features are selected from a fully detailed map on the fly and sent to the user.

Such a system has presented a multimodal approach for both explicit and implicit user profiling. Each individual user profile is created and maintained for map personalisation. The more frequently a user uses the system, the more accurately the user profile is able to represent the user's preferences, offering better map personalisation. However, the system still lacks efficiency for new or occasional users, having no user profile stored.

Also this system is restricted for use with either Tablet PCs or advanced PDAs equipped with a stylus and speech instruction recognition facilities, to interact with map features. Implementing speech recognition needs the first time user to complete a "getting to know the user" speech training session, wherein the user has to spend a considerable amount of time reading back specific sentences in order that the system gets used to the user's pronunciation. After the "getting to know the user" training session, the verbal instructions can be normally recognised at an acceptable level of accuracy. However, this system does not address the major issues related to speech recognition, such as, i) the user accents and ii) background noises especially from multiple users present. The system currently offers map personalisation for regular users, who have a user profile stored, only. It does not support the personalisation of maps for new/occasional users who are accessing the system for the first time, or who do not wish to store a user profile with the system.

Some web-based systems have also been developed that monitor users' implicit interactions and record this data for further analysis. A few of these systems will be discussed to examine their feasibility for incorporation into providing map personalisation and spatial feature recommendation. In [37] Goecks and Shavlik developed a system wherein, a user's preferences are collected implicitly by unobtrusively monitoring the user's behaviour while browsing pages. Particular normal user actions on web pages, such as the number of hyperlinks clicked on a page, or the amount of scrolling performed on the page, are used as a "surrogate" for the measurement of the user's interest level in that page. It is assumed however, that a user's normal actions on web pages, as discussed above, present a strong relationship with the user's preferences or interests. In this thesis, a user's spatial feature preferences are collected by recording the enabling/disabling of particular spatial features of interest on the displayed map.

The analysis of users' mouse movements and activity can be another means of determining a user's content preferences. The Cheese system [25] furthers the system developed by Goecks and Shavlik [37] by recording all mouse movements, instead of only mouse clicks. It monitors the position of the mouse and time spent at each position, recording this data for further analysis to determine users' interests. This system proves to be effective for web page content providers in increasing the effectiveness of their interface, through the analysis of certain mouse behaviours such as, hesitation on links/text before clicking, and intentionality or fast clicking of familiar links. However, in such a system incorrect information can be unintentionally attained through users leaving their mouse on certain links while focusing on another part of the web page.

Certain user interactions with content can reveal important information about the user's interests in that particular content. Kelly and Belkin [19] explore the techniques of ascertaining a user's interest implicitly, based on three hypotheses; i) users will spend more time reading documents they find relevant, ii) users will scroll through the page more often in documents they find relevant, and iii) users will have more interaction with the documents they find relevant. Using these hypotheses, we can develop a map personalisation algorithm to retrieve and present spatial features of possible interest to a target user, by analysing the spatial features most commonly interacted with by the majority of users

2.3 MAP PERSONALISATION

In this thesis, personalisation of maps is the process of extracting particular geo-spatial information/services from detailed datasets tailored to the user, by means of analysing their preferences stored in the user profiles.

Two main information filtering techniques currently exist regarding map personalisation, namely content-based filtering and collaborative filtering. Content-based filtering [71] is used for regular users who have their user profile stored. This method provides detailed information filtering through contents, assuming the user has created and stored his/her user profile. These user profiles are continually updated with users' current preferences or interests, through their explicit and implicit interaction with the developed interface. However, it does not offer any map personalisation for new/occasional users, who do not wish to have their user profile stored. On the other hand, collaborative filtering techniques [11, 41] help address this issue of offering map personalisation to new/occasional users, by determining the probability of similarity between the new user and other existing users, based on their interactions with map features. It utilises the opinions of like-minded users to make an inference for a target user. This filtering approach is highly advantageous in a system consisting of a great number of users, as well as one that contains large datasets [69]. Without collecting and administrating enormous numbers of individual user profiles, collaborative filtering approaches dramatically improve the efficiency of filtering large amounts of geo-spatial information. However, this approach lacks the detailed filtering of spatial information provided through content-based filtering.

Therefore, a combined approach to map personalisation, implementing content-based filtering for regular users, and collaborative filtering for new/occasional users, can be developed [62] to provide an efficient method of offering personalised maps/geo-spatial services to users. Outlined below is some recent research conducted utilising content-based filtering techniques to provide personalised maps to end users.

Weakliam et al. [43, 44] have proposed a user model to profile a user's preferences from all implicit map interactions, monitored and stored in an XML (Extensible

Markup Language) file. With the user's map interactions from previous sessions, map features and map regions of interest can be inferred for each individual user and gradually refined by frequent uses. Furthermore, association rule mining[2] has been adapted to cluster commonly liked map features and improve map feature recommendation.

Such a user model has been implemented into a working system, where relevant map features are selected for each individual user based on their user profiles. An initial system evaluation in terms of user profile acquisition and similar map feature clustering has shown some promising results.

However, map personalisation is manipulated at feature types only (e.g., streets, highways, parks) rather than individual features. The map personalisation is compromised by the coarse level classification of features. Meanwhile, this limits the item sets (groups of features) available in the system for association rule mining. Therefore, this challenges the resultant association rules for map features clustering, as an association rule needs the input and support of several hundreds of item sets before it can be considered statistically significant.

Yang and Claramunt [83, 84] developed a method that applies k-order Markov chains[3] to provide personalisation regarding spatial entities embedded in a web document. The prototype experiment is applied to historical and sightseeing places in the city of Kyoto, modelled as spatial entities containing semantic attributes and displayed as images or symbols on a web map interface. The Markov chains combined with spatial proximity (contextual form of inverse distance between two spatial entities), semantic similarity (degree of similarity of two spatial entities in the semantic domain), and a reinforcement process based on implicit users' relevance feedbacks (negative or positive) to personalised presentations, provide a framework to model users' navigation trails and predict future user interactions with the spatial information. Users' navigational trails and behaviour are recorded using historical web logs that store the page-views requested by the user i.e., the number of times

[2] A popular and well researched data mining technique used to find similarities or interesting relations between items in large databases.

[3] Yang and Claramunt [84] state that Markov chains have been extensively used in the past for the purpose of predicting the next state of a system, given a sequence of previous states.

certain spatial entities or links are clicked or requested. Markov chains are basically used to predict the next state based on a sequence of past states of the system. Deshpande and Karypis [49] state that Markov chains of low order are unable to efficiently predict the next state of the system as there are not enough past states, whereas, high order chains result in a high number of past states, which increases their space and run-time requirements. This system utilises third order Markov chains for effective personalisation. The probability of progressing from one spatial entity/page-view to the next is statistically calculated, by the Markov chain model, from the web logs used to store users' historical navigation trails.

Using Markov chains of the right order for web personalisation proves to be effective in predicting users' future navigation actions based on their previous navigation trails. Employing a reinforcement process, implicitly recording the user's satisfaction or dissatisfaction with the personalised result, provides a better understanding of the user's navigation goals or preferences. According to Winkelholz and Schlick [17], the use of statistical methods to decide which states are accepted for the model is useful in two ways; i) the prediction quality of future states is better than conventional algorithms, and ii) each state has a more significant meaning. However, the system design is currently just conceptual.

The framework of the Markov chain model has been designed, but not implemented into a working system. Plans are in place to include web-based travel planning and personalising the user's travel experience based on past navigation experiences. However, this would be better suited in mobile environments, where the user is on the move, instead of a desktop PC. However, using Markov chains often takes up quite a lot of time and memory in finding repetitive previous states in the process of generating the next states [64], which would need to be addressed, if implemented in a mobile environment.

Jabeur & Moulin [58] developed a system based on pre-computed and cached vector map data being sent across to the end user in the form of maps. This approach makes use of the techniques of multiagent systems and cartographic generalisation to deliver the maps to its users. The spatial objects are categorised into layers of interest, based on the user's spatial interests. Spatial objects of similar interest level to the user are grouped into one layer of interest. Therefore, upon a user's request, layers of the map with features of interest to the user can be generated from the pre-computed vector map data on-the-fly and sent to the user. Therefore, there exist multiple layers at different levels of preference to the user. The layer of utmost importance is initially sent to the user, followed by the other layers, according to their level of interest to the user.

One of the advantages of this system is the ability to produce multiple representations of a spatial object, such as geometric, semantic and graphic representations. This can be set so that when the scale of the map is quite large an object could be represented as just a geometrical location and when the scale is relatively small a graphic representation of the object could be used for better visualisation of the object. By grouping features of similar interest to the user into layers, allows the sending of data most relevant to the user first, followed by additional data or less relevance, allowing the user to stop the transmission of layers at any time, as he/she receives the desired spatial content.

This system, however, explicitly registers information about users' preferences by asking them to provide details about their spatial feature preferences. The speed of map generation and data transfer needs to be addressed further to minimise the time taken to deliver maps to the user.

The following system utilises collaborative filtering techniques by creating user models for different types of users, and inferring their preferences based on their implicit and explicit interactions with the map features. The TArcHNA [21] system offers the ability to link archaeological findings to particular geographic burial tomb locations. The system aims to offer the user spatial and non-spatial (text, pictures and movies) data, regarding archaeological information in the city of Tarquinia, tailored to their own personal needs. This system can be accessed from desktop PCs, PDAs, or smart phones. The architecture consists of a remote server, storing the spatial and non-spatial archaeological data that is requested by the user, and a modelling component, that carries out the parsing of the data and forms a communication link between the data server and user interfaces. Personalisation in this system is proposed to be achieved by capturing the user's implicit interactions with the data, such as mouse movements, mouse clicks, zooming and panning. Explicit user interactions such as searching or choosing objects of map are also considered. Implicit and explicit interactions are used to create user models for different groups of users such as tourists, archaeologists and historians. For the mobile device application it preloads the datasets from the database onto the device, the contents of which are determined by the information stored in the user profile.

One of the drawbacks of this system is that, due to the datasets being preloaded on the mobile device, based on the preferences in the user's profile, the user may not be able to access certain needed data that the user profile may not have considered relevant for the user. This system is also limited to PDAs or smart phones, making it unfeasible for everyday mobile device users to access the system. Also the implementation of personalisation proposed is solely conceptual and under development.

Another system that uses collaborative filtering techniques to retrieve multiple spatial features has been implemented in [4]. It makes use of Bayesian Inference[4] networks to rank spatial themes of relevance to a user based on some available historical data. It focuses on the automatic creation of a Bayesian network through Bayesian learning algorithms, based on some present historical data. This automatically created Bayesian network can then be used to retrieve relevant spatial themes, based on a test set of user queries. Spatial relationships between various GIS themes are found and converted into casual spatial relationships so that they can be incorporated into the Bayesian learning algorithms. The spatial relationships between GIS themes vary in strength depending on their spatial proximity between each other. A few Bayesian learning algorithms incorporating spatial relationships are then evaluated and compared in terms of providing the most efficient and improved GIS theme retrieval performance.

By automatically creating a Bayesian Inference network, this system eliminates the need for users to manually select GIS themes to be displayed; instead it uses historical data to automatically select GIS themes with high rankings. This is useful in systems

[4] "Bayesian inference is a technique for estimating unobservable quantities from observable quantities" [55].

where a lot of themes are present, and the user is not always sure about the selection of themes that would be of interest to him/her.

The disadvantage of implementing such a Bayesian network is the time taken to rank the GIS themes based on their spatial relationship and historical data. Through the evaluation, the time taken to create a Bayesian network for geo-spatial data consisting of 30 GIS themes was about 250 seconds. This paper does not provide any technique for recommending GIS themes to registered users. Collecting data from registered users would be beneficial in attaining more data to determine similarities in various GIS themes.

2.4 OUR PROPOSALS

In this thesis, some of the major issues related to user profile acquisition and map personalisation have been addressed, such as i) information overload on the small screens of everyday mobile devices, ii) large amounts of geo-spatial data being sent across mobile networks and the Internet, resulting in unsatisfactory download time and excessive download costs, and iii) lack of an efficient method of map personalisation for retrieving spatial features of significant relevance to a target user.

We propose a combined map personalisation approach, incorporating content based filtering as well as collaborative filtering techniques. Content-based filtering works well for retrieving spatial features for regular users who have their user profile stored on the server, and collaborative filtering can adequately retrieve relevant spatial features for new/occasional users having no user profile present. Inference of a user's spatial feature preferences through content-based filtering is made possible by analysing their interaction with particular spatial features of interest on the map, such as motorways, petrol stations, shopping malls, etc. This information is then stored in user profiles that are constantly updated with the user's current spatial feature preferences. For collaborative filtering, users are grouped into three user categories (tourists, surveyors, and local residents), and personalised maps are retrieved based on the preferences of other existing like-minded users.

An advanced inference technique of association rule mining can be used to offer map personalisation through collaborative filtering, by determining the spatial features most commonly retrieved together by users belonging to the same user category. For instance, when a user does not have an existing user profile, the spatial features of interest to the target user can be automatically inferred, by analysing the spatial features most commonly retrieved by like-minded users. Whereas, in the case of a new user with no current user profile, support values for each spatial feature can be automatically calculated and stored in their user profile, based on the spatial feature support values of other like-minded users. The calculation of these support values can be performed by various advanced inference techniques, such as Bayesian inference, association rule mining, Markov chains, etc.

This combined map personalisation algorithm is then implemented into a working system, based on a four tier, distributed client-server architecture, using open and standard system development tools. It consists of a database server, application server, and mobile users. The database server hosts the real world geo-spatial datasets and the user profiles. The application server is used to deploy the GUI and map rendering application, which can then be accessed by mobile users across mobile networks and the Internet. Users can use their everyday mobile devices to access this system anytime, and from anywhere, provided they possess access to a valid Internet connection. The GUI developed has been programmed in Java, offering platform independency. This enables our system to be accessed by most everyday mobile devices. Also, our system uses standard communication protocols, such as WAP and HTTP, as means of communication between the various tiers in the system architecture. This allows our system to be interoperable with other mobile GIS, without the need for excessive modifications.

CHAPTER - 3

COMBINED MAP PERSONALISATION ALGORITHM

3.1 INTRODUCTION

Maps/geo-spatial services personalised based on individual user profiles, has been acknowledged as a viable solution to some of the existing issues related to mobile GIS. Two information filtering techniques called content-based filtering and collaborative filtering have been proposed to offer map personalisation for regular users, as well as new or occasional users respectively. Regular users are users who have their user profile stored and use a mobile GIS frequently, whereas occasional users are users who do not have a current user profile stored, neither are they interested in creating one since they may be one-off or infrequent users. New users are users interested in creating their user profile, not having any current user profile stored.

For content-based filtering, individual user profiles are required in order to personalise content based on a user's interest [30] in particular geo-spatial features of a map. These user profiles can be collected by explicitly requesting a user's spatial feature preferences, and/or by implicitly and unobtrusively monitoring a user's interaction with particular spatial features and the map interface. This information can then be analysed and interpreted to infer a user's preferences regarding particular geo-spatial features to a satisfactory degree of accuracy. However, this technique is not applicable for new or occasional users without having any user profile stored. Collaborative filtering has been proposed for new/occasional users who do not have a user profile stored. It utilises the opinions or preferences of like-minded users to make an inference for a target user. To offer collaborative filtering, association rule mining, an advanced inference technique, can be used, by determining the spatial features most commonly retrieved by similar or like-minded users.. However, this approach lacks the detailed filtering of spatial information provided through content-based filtering.

Therefore, there is need to propose a combined spatial feature personalisation approach, accommodating for regular users, as well as new/occasional users. Such an algorithmic approach would incorporate content-based filtering for regular users, and collaborative filtering for new/occasional users, producing an effective and efficient method of delivering personalised maps/geo-spatial services to users, due to more users with varied interests being able to access these maps/geo-spatial services.

The remainder of this chapter is structured as follows. Section 3.2 presents the user profile acquisition methods used in our combined algorithm. Section 3.3 will discuss the content-based information filtering technique in detail, describing its implementation and methodology. In Section 3.4, the collaborative filtering approach is explained, in conjunction with association rule mining. Section 3.5 defines the combination of these two information filtering techniques detailed for map personalisation. Finally, a conclusion is outlined in Section 3.6.

3.2 ACQUISITION OF USER PROFILES

One of the essential elements of delivering personalised maps to users is the effective collection of a user's preferences or interests regarding spatial features (main roads, shopping malls, highways, etc.), to be stored into user profiles. The collection of such user profiles can be performed by explicitly requesting spatial feature preference information from the user, and/or implicitly capturing users' spatial feature preferences through their interactions with the developed interface.

In this research, we use implicit user profile acquisition methods to store users' spatial feature preference information. A user's preferences regarding particular spatial features are recorded and updated on a session by session basis. One "session" is defined as the time between when the user logs in until the user logs off. At the end of each session, the spatial features enabled on the map are recorded in a transactions table as one "transaction". A user's previous and current transactions can then be analysed to determine their interest in particular spatial features, stored as "support" values for each feature in their user profile. For instance, when a user disables the display of a particular spatial feature on the map, such as "Petrol Stations", the reasonable assumption is made that the user is not interested in viewing the locations of petrol stations. Therefore, the next time the user logs in, the spatial feature "Petrol Stations" would not be given preference for display on the map.

Once these user profiles have been collected, maps/geo-spatial services can be personalised for users, based on the information stored in the user profiles. In this thesis, two major information filtering techniques, content-based filtering and collaborative filtering, have been researched and investigated to determine their feasibility in being implemented for map personalisation.

3.3 CONTENT-BASED FILTERING

Content-based information filtering deals with the selection of specific items from a large detailed collection, that a user is likely to be interested in or find useful. In GIS content-based filtering is mainly used for the purpose of extracting geo-spatial features/services from a fully detailed map/service, based on a particular user's preferences or requirements stored in their user profile. Therefore, this method works well to tailor the generalised map/service to the individual user's spatial feature preferences or requirements.

This information filtering technique analyses the information stored in user profiles to retrieve spatial features which would be of most relevance to the individual user. In this thesis, content-based filtering has been proposed for users who agree to have their own individual user profile stored in a database. Upon creation of such a user profile, a few details, such as the user's login information, and a user category best describing the user (tourist, surveyor or local resident), can be stored into a particular table located in a database. "Support" values (values between 0 and 1) associated with each feature type (such as motorways, petrol stations, post shops etc.) can be calculated and stored in the user profile, based on the spatial feature support values stored in user profiles of other users belonging to the same user category. A support value of 1 could indicate greatest interest in the feature, whereas a support value of 0 could indicate no interest at all in that particular feature. These support values, associated with each feature type, are constantly updated after every session of usage, based on current and previous spatial feature transactions of that user. The next time the user logs in, a personalised map containing only spatial features with support values greater than a specified threshold value, will be displayed.

Recording different users' transactions in each map session, i.e., the spatial features retrieved and displayed at the end of each session, makes it possible to provide users with a personalised map based on their spatial feature preferences. These transactions could then be used to assign new support values for each feature stored in the user's profile. Each spatial feature's frequency of occurrence can be checked and compared to other spatial features. The most recent transactions existing for that user are then used, in order to provide support values based on the user's current preferences and requirements. Based on the frequency of occurrence of a particular spatial feature f, its support value S stored in the user profile is calculated and updated as:

$$S = \frac{f}{T}$$

where, T is the total number of transactions stored for that user. For instance, referring to Table 3.1, to calculate the support value of the spatial feature "Petrol Stations" for a particular user with User ID 5, the frequency of occurrence of the spatial feature is checked, which, as we can see from Table 3.1, is six times in the last ten transactions.

Table 3.1 - Example User Transactions

Spatial Feature	User ID	Total Transactions	Frequency of Occurrence
Petrol Stations	5	10	6

Therefore, the support value for the spatial feature "Petrol Stations" is:

$$S(petrol) = \frac{f(petrol)}{T} \quad \rightarrow \quad S(petrol) = \frac{6}{10} = 0.6$$

This process can be repeated after each user session, updating the feature support values stored in the user's profile. Therefore, allowing us to gain a better perception of the user's possibly changing preferences and requirements, regarding spatial feature content retrieval.

However, it is impractical for content-based filtering approaches to provide relevant spatial feature content retrieval for new/occasional users, who do not have an existing user profile stored. Therefore, it is acceptable to propose a personalisation approach to accommodate these new/occasional users. A way of achieving this is through the implementation of collaborative filtering techniques as discussed in the following section.

3.4 COLLABORATIVE FILTERING

The basic idea of collaborative filtering based algorithms is to provide recommendations or predictions about items, based on the opinions or preferences of other like-minded users [11].

Collaborative filtering operates by clustering similar users into a group, and making an inference for a target user's preferences, based on the preferences of similar existing users. This technique of information filtering is largely implemented for occasional users who would not want to be bothered with storing their user profile, or for new users who wish to create a new user profile. Without administrating enormous individual users' profiles, collaborative filtering approaches dramatically improve the efficiency of filtering large amounts of geo-spatial information.

With the use of data mining techniques, such as association rule mining, Bayesian networks, Markov chains, etc. collaborative information filtering can be achieved.

Bayesian networks require a substantial amount of time to build learning models based on a training set of data, while at the same time being computationally quite complex. Hence, they are practical for environments where user preferences do not change or change slowly, due to the large amounts of time taken to build the learning models [31]. Also, Bayesian networks creation is often non-trivial and requires a relatively large amount of effort to implement. Furthermore, Bayesian networks have been found to be impractical for large systems with a lot of variables [14]. Markov chains have also been found to be computationally expensive, resulting in excessively long solution times, and requiring a large size of initial training data to provide resultant predictions [51]. In this thesis, we need to accommodate map personalisation for users' changing spatial feature preferences, and store relatively large amounts of users' transactions in a database. Therefore, Bayesian networks and Markov chains appear to be unsuitable for our field of application.

Association rule data mining, on the other hand, is one of the most significant and well researched large data mining techniques. It's computational time and complexity is quite satisfactory, compared to Bayesian networks or Markov chains, which enables the implementation of map personalisation based on users' continually changing preferences or interests. Therefore, association rule mining can be satisfactorily implemented in order to make an inference of a target user's spatial feature preferences, based on the preferences of other like-minded users. One of the main reasons for implementing association rule mining is the fact that it is probably the most elementary data mining technique, as well as the most widely used [24]. Most importantly, association rule mining has been extensively used as a collaborative filtering method, offering personalisation of content, based on collected information regarding similarities of preferences between users [8, 40, 80].

In the following section, association rule mining will be investigated for its feasibility in the retrieval of relevant spatial features for a target user with no existing user profile.

3.4.1 ASSOCIATION RULE DATA MINING

The main purpose of association rule mining is to discover interesting and informative patterns in a transaction database. One of the earliest forms of association rule mining was the use of market basket analysis, proposed by Agrawal et al. [68]. It aims to discover shopping patterns of customers (i.e., products commonly purchased together) based on previous purchase transactions. For instance, the association rule $\{bread, milk\} \Rightarrow \{cereal\}$ found in a supermarket's sales data would indicate that if a customer buys bread and milk together, they are likely to buy cereal as well. This information proves to be quite useful in making decisions about marketing activities, such as, promotional pricing or product placements in supermarkets.

The traditional application of mining association rules is market basket analysis [66, 67, 68]. However, since then association rule mining has been used in a wide variety of applications, including census data, insurance data, medical diagnosis, causes of plan failures, web personalisation, text data, and publication databases [9, 13, 18, 50, 54, 70, 73].

Association rule mining is originally defined as follows:
Let $I = \{i_1, i_2, \dots, i_m\}$ be a set of items, referred to as an itemset.
Let $T = \{t_1, t_2, \dots, t_n\}$ be a set of transactions, known as the transaction database. Each transaction in T contains a subset of the items in itemset I. An association rule expresses an implication of the form $X \Rightarrow Y$, where $X, Y \subseteq I$ are itemsets and $X \cap Y = \emptyset$. X is referred to as the antecedent, and Y, the consequent of the rule.

To determine frequent association rules, two important basic measures for association rules are present, support and confidence. The support of an association rule is defined as the percentage/proportion of transactions in the transaction database that contain the itemset $X \cup Y$, such as:

Equation 3.1 - Support of an Association Rule

$$supp(X \cup Y) = \frac{\sum(X \cup Y)}{T}$$

where, T = total transactions in database. The confidence of an association rule is defined as the percentage/proportion of the transactions containing $X \cup Y$ to the percentage/proportion of transactions containing X, such as:

Equation 3.2 - Confidence of an Association Rule

$$conf(X \Rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)}$$

The support of an association rule determines the statistical significance of the rule, whereas, the confidence of a rule is a measure of its strength. The minimum threshold values of these two measures; support and confidence, can be adjusted accordingly to filter out association rules that are not quite useful or interesting. The generation of association rules requires the processing of two major steps. Firstly, the minimum support value is applied to all itemsets, to find the frequent itemsets in the database. And secondly, those frequent itemsets, along with the minimum confidence value are used to form the association rules.

Recently, association rule mining has been applied to commercial collaborative recommendation systems, such as Amazon.com [26, 76, 85] and eBay [76, 85]. Amazon.com is able to recommend items that might be of interest to a user, by analysing three aspects in a user profile, such as items purchased through their website, items owned through non-website purchases, and items rated by the user. This data is compared with other users to make recommendations of items that might interest the current user. For instance, Amazon.com discovers that customers who buy a particular book, *Book A*, also buy another book, *Book B*, together. Thus they can display this information as a package to all potential buyers of either book. Sandvig et al. [40] shows how association rule mining is a much more robust approach as compared to other memory-based collaborative filtering algorithms, such as K-nearest neighbour. This is another crucial factor as it reduces the load on the server to process incoming requests and transmit responses, enabling smooth operation and feasibility for the adaptation of this approach of collaborative filtering to the context of maps and geo-spatial features/services.

A large number of algorithms exist to find frequent itemsets in a database, based on a minimum support threshold value. Apriori [68], Eclat [53], and FP-Growth [38] are some well known algorithms present at the moment. All algorithms for finding association rules mostly result in the same set of generated association rules. Differences only exist in the processes implemented, their computational efficiencies and memory requirements. The Apriori algorithm is currently the best known algorithm to mine association rules from a database containing transactions [6]. Implementing the Apriori algorithm to find association rules for groups of commonly like map features/services enables the use of association rule mining to retrieve spatial features of probable interest to a user, providing map personalisation. Apriori uses an iterative “bottom-up” approach, in which frequent subsets are extended, one item at a time (also known as candidate generation), and groups of candidates are tested and analysed against the data. The following section provides an overview of the Apriori algorithm for association rule mining along with the execution process and computation time.

3.4.2 APRIORI ALGORITHM SUMMARY AND PSEUDO CODE

The Apriori algorithm is an efficient iterative association rule mining algorithm developed by Agrawal et al. [66], in which items that occur frequently together are found. Apriori uses the Breadth First Search (BFS)[5] tree search algorithm and employs a hash tree structure to total the number of candidate itemsets. It is used to generate *k* length candidate itemsets from *k-1* length candidate itemsets. The candidate itemsets having infrequent subsets are then pruned. Candidate itemsets are determined in order of their size, and for each size level, the algorithm examines whether they are frequent or not. Initially, the algorithm finds the frequent 1-itemsets, based on the minimum support value specified, and generates candidate itemsets for the second level. The support values for these second level

[5] Breadth First Search is a basic graph exploration technique [46] that systematically explores the edges of a graph, to discover every vertex reachable from the source vertex [32].

candidate itemsets are determined by scanning through their frequency in the transactions database, and those candidate itemsets that meet the minimum support value are used to generate the candidate itemsets for the next level (level 3). The Apriori algorithm continues in this fashion, constantly iterating up through the levels. At each level k, the frequency of the candidate itemsets C_k is checked, and if $support(C_k) > minimum\_support$, it is used to generate candidate itemsets for the next level C_{k+1}. When the algorithm reaches the level k, where there exists no candidate itemset C_k whose support value is greater than the minimum support value, the algorithm comes to a stop. The characteristic of the Apriori algorithm is that on each level, it has the knowledge of the frequent itemsets from the previous level, which implies it has knowledge of frequent itemsets from all the past levels. Using this information, candidate itemsets can be generated on the current level, and the infrequent candidate itemsets can be pruned. In general, a transaction database that contains k items will potentially be able to generate up to $2^k - 1$ frequent itemsets, excluding the null set [63].

Some of the key concepts of the Apriori algorithm are listed below:

- **Frequent Itemsets:** Finding all sets of items that have a support value greater than the minimum support value
- **Apriori Property:** All subsets of the frequent itemsets must be frequent as well, i.e., if $\{A, B\}$ is frequent, then $\{A\}$ and $\{B\}$ are frequent itemsets as well

The main steps of the iteration are detailed as follows:

Defining:

C_k as a candidate itemset of size k

L_k as a frequent itemset of size k

Step 1) Find frequent itemset L_{k-1}

Step 2) Join step: C_k is generated by joining L_{k-1} with itself (Cartesian product $L_{k-1} \times L_{k-1}$)

Step 3) Prune step (Apriori property): Any itemset of size $(k-1)$ that is not frequent cannot be a subset of a frequent itemset of size k, hence it should be removed

Step 4) The frequent itemset L_k has been determined

The pseudo code for the Apriori algorithm is briefly outlined below.

C_k: Candidate itemset of size k
L_k : frequent itemset of size k
min_sup: minimum support value
L_1 = {frequent 1-itemsets}

for (k = 1; L_k != ∅; k++) **do begin**
    C_{k+1} = candidates generated from L_k;
    for each transaction t in database **do**
        increment the count of all candidates in C_{k+1} that are contained in t
    L_{k+1} = candidates in C_{k+1} with support > *min_sup*
end
return $\cup_k L_k$;

The computation time is an important factor for any algorithm, and determines it efficiency and effectiveness to run in real time. The computation time for the Apriori algorithm to find all frequent itemsets in a specified transaction database varies with the number of items or candidates available, and the number of transactions stored in the transaction database. Another major factor affecting the computation time for the algorithm, is the specification of the minimum support value assigned to prune the candidate itemsets. Lower support values could increase the levels the algorithm iterates through to search for candidate and frequent itemsets. The Apriori algorithm is relatively fast in comparison with other present data mining algorithms, due to its property stating that if at least one subset of a candidate itemset is infrequent, the candidate itemset must be infrequent as well. This eliminates the need to iterate through non-frequent itemsets. In certain circumstances, all association rules can be found in linear time, or $O(n)$ time in the 'big O notation[6]'.

The Apriori does suffer from some complexity issues, such that its efficiency is inadequate in the presence of a large number of frequent itemsets, or the length of the longest frequent itemsets is large. In [3] the Input/Output (I/O) complexity of the Apriori algorithm has been calculated as follows. Lets define T as the size of the input transactional database, and B as the block size or the maximum number of candidate itemsets ($2^k - 1$) that can be transferred between disk and Random Access Memory (RAM). If a computer's RAM is capable of containing all intermediate data produced by the algorithm (i.e., candidate and frequent itemsets), then the I/O complexity of the Apriori algorithm is:

Equation 3.3 - I/O complexity of Apriori

$$I/O\ complexity = k_{max} \times \frac{T}{B}$$

where, k_{max} = length of the longest itemset, T = size of the transaction database, and B = block size. However, if the algorithm's intermediate data does not fit in the RAM, the I/O complexity of the algorithm would end up being a function of the sizes of the candidate and frequent itemsets L_k and C_k as well.

3.4.3 APRIORI IMPLEMENTATION FOR MAP PERSONALISATION

In this section, the implementation of the Apriori algorithm of association rule mining to provide relevant spatial features for new/occasional users, based on the preferences of existing like-minded users, will be discussed and examined for its feasibility. In our thesis, association rule mining has been implemented as follows.

The number of user sessions, and the spatial features contained in those sessions, provides the transactions required for the Apriori algorithm. The algorithm iterates through the transactions and computes association rules for groups of commonly liked map features/services by the user categories. Initially, the resultant frequent

[6] The big O notation for functions has been widely used in the analysis of algorithms to determine the computational efficiency of algorithms [72].

itemsets are derived and stored in the Frequent Itemset Graph[7] (Figure 3.4). The graph is constructed into levels from 0 to *k*, where the single root node at level 0 corresponds to an empty itemset and *k* is the maximum size among all frequent itemsets. Each node at depth *d* corresponds to an itemset *I* of size *d* and is linked to itemsets of size $d + 1$ that contain *I*. Each node also stores the corresponding support value.

The following is an indicative example of the processes included in the Apriori algorithm, in terms of geo-spatial features requested by a user. Table 3.2 is a transaction table for a particular user, showing the spatial features requested in each transaction. The spatial features are denoted by F1 to F8.

Table 3.2 - Example User Session Transactions

Transaction Number	Features Present
1	Malls (F1), Petrol Stations (F2), Motorways (F3), Airports (F4), Highways (F5)
2	Petrol Stations (F2), Motorways (F3)
3	Malls (F1), Petrol Stations (F2), Post Shops (F6)
4	Petrol Stations (F2), Malls (F1), Ambulances (F7)
5	Malls (F1), Post Shops (F6), Streets (F8)

Using a minimum support value, *min_sup*, of 40% or 0.4, all frequent itemsets need to be found using the Apriori algorithm. Initially, the transaction database (Table 3.2) is scanned to get the support *S* of each 1-itemset. *S* is then compared with *min_sup*, and a set of frequent 1-itemsets, L_1, are found as illustrated in Figure 3.1, and presented at "Depth 1" in Figure 3.4.

[7] A directed acyclic graph used to represent the itemsets stored in the database.

Candidate Itemsets, C_1 → **Frequent Itemsets, L_1**

Feature ID	Feature	Support
F1	Malls	4/5
F2	Petrol Stations	4/5
F3	Motorways	2/5
F4	Airports	1/5
F5	Highways	1/5
F6	Post Shops	2/5
F7	Ambulances	1/5
F8	Streets	1/5

Feature ID	Feature	Support
F1	Malls	4/5
F2	Petrol Stations	4/5
F3	Motorways	2/5
F6	Post Shops	2/5

Legend: ▭ Itemsets pruned due to its support value < *min_sup* value (0.4)

Figure 3.1 - Finding Frequent 1-Itemsets

To find the frequent 2-itemsets, we then, use L_{k-1} (L_1) join L_{k-1} (L_1) to generate a set of candidate *k*-itemsets (2-itemsets), and use Apriori property to prune the unfrequented *k*-itemsets (2-itemsets) from this set (Figure 3.2). Apriori property states that if an item *A* is added to the itemset *I*, then the resulting itemset (i.e., $I \cup A$) cannot occur more frequently than *I*. Therefore, $I \cup A$ is not frequent either, i.e., $P(I \cup A) < min\_sup$.

Candidate Itemsets, C_2 → **Frequent Itemsets, L_2**

Feature ID	Feature	Support
{F1, F2}	Malls, Petrol Stations	3/5
{F1, F3}	Malls, Motorways	1/5
{F1, F6}	Malls, Post Shops	2/5
{F2, F3}	Petrol Stations, Motorways	2/5
{F2, F6}	Petrol Stations, Post Shops	1/5
{F3, F6}	Motorways, Post Shops	0

Feature ID	Feature	Support
{F1, F2}	Malls, Petrol Stations	3/5
{F1, F6}	Malls, Post Shops	2/5
{F2, F3}	Petrol Stations, Motorways	2/5

Legend: ▭ Itemsets pruned due to its support value < *min_sup* value (0.4)

Figure 3.2 - Finding Frequent 2-Itemsets

This process is then constantly iterated through until all candidate *k*-itemsets' support values are less than the minimum support value, leaving no further frequent itemsets to be generated, as shown in Figure 3.3.

Candidate Itemsets, C_3

Feature ID	Feature	Support
{F1, F2, F3}	Malls, Petrol Stations, Motorways	1/5
{F1, F2, F6}	Malls, Petrol Stations, Post Shops	1/5
{F1, F3, F6}	Malls, Motorways, Post Shops	0
{F2, F3, F6}	Petrol Stations, Motorways, Post Shops	0

L_3 = NULL

Legend: Itemsets pruned due to its support value < *min_sup* value (0.4)

Figure 3.3 - Finding Frequent 3-Itemsets

Once there exists no further frequent itemsets, the Apriori algorithm stores all the discovered frequent itemsets into the Frequent Itemset Graph (shown in Figure 3.4), from where the frequent itemsets can be easily accessed and selected.

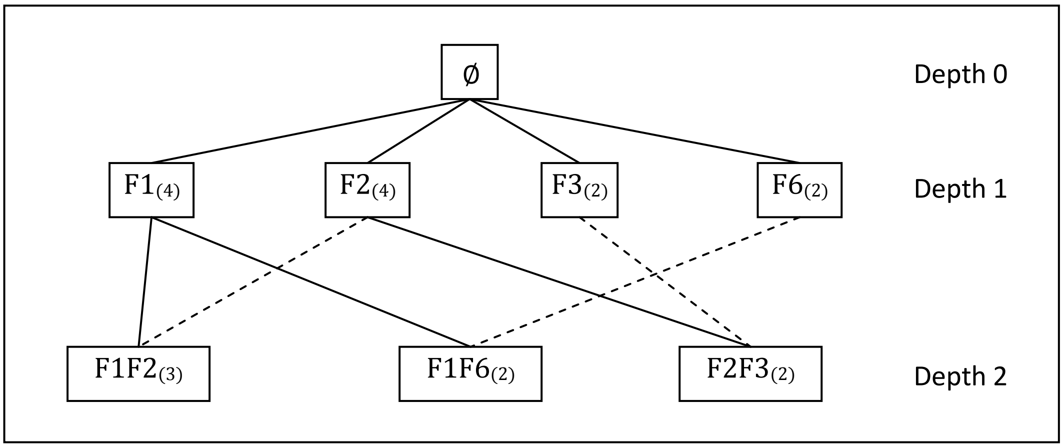


Figure 3.4 - Frequent Itemset Graph

The next step requires the generation of all non-empty subsets of *I* for each frequent itemset-*I*, keeping the *min_sup* at 40% and setting the minimum confidence value, *min_conf* at 70%. Table 3.3 shows the results generated from this step.

Table 3.3 – Non-Empty Subsets of I

Feature ID	Feature	Support (A∪B)	Support (A)	Confidence (A∪B)
{F1, F2}	Malls, Petrol Stations	60%	80%	75%
{F1, F6}	Malls, Post Shops	40%	80%	50%
{F2, F3}	Petrol Stations, Motorways	40%	80%	50%
{F2, F1}	Petrol Stations, Malls	60%	80%	75%
{F6, F1}	Post Shops, Malls	40%	40%	100%
{F3, F2}	Motorways, Petrol Stations	40%	40%	100%

Legend: ▢ Items to be Pruned due to its confidence value < *min_conf* (0.7)

And finally, once the frequent itemsets are obtained, association rules can be generated for every non-empty subset s of I, such as "$s \Rightarrow (I - s)$" if:

$$\frac{support\_count(I)}{support\_count(s)} \geq min\_conf$$

where, *min_conf* is the minimum confidence threshold set at 70%. The resultant association rules for all the non-empty subsets in Table 3.3 are presented in Table 3.4.

Table 3.4 - Generated Association Rules

	Strong Rules	Support (A∪B)	Confidence (A∪B)
{F1}⇒{F2}	Malls ⇒ Petrol Stations	60%	75%
{F2}⇒{F1}	Petrol Stations ⇒ Malls	60%	75%
{F6}⇒{F1}	Post Shops ⇒ Malls	40%	100%
{F3}⇒{F2}	Motorways ⇒ Petrol Stations	40%	100%

3.5 THE COMBINED MAP PERSONALISATION APPROACH

Combining content-based and collaborative filtering techniques would result in an effective model for map personalisation, accommodating for regular users as well as new/occasional users, due to more users with varied interests being able to access the personalised maps. In the case of a new user, they can be offered the option of either, creating and storing a user profile for themselves, or simply selecting their user category. If they do store their user profile, collaborative filtering techniques are used to present the user with an initial personalised map, following which, content-based filtering can be implemented for map personalisation, based on the preferences collected in their user profile. On the other hand, if a user chooses not to store their user profile, again collaborative filtering can be utilised, where a user can choose his/her user category type, and personalised maps are retrieved based on the spatial feature preferences of other similar users.

Combining content-based and collaborative filtering techniques to offer map personalisation would minimise the drawbacks or disadvantages currently existing with the independent implementation of either of these information filtering techniques. The pseudo code of such a combined approach for map personalisation follows.

```
IF a new user {
   IF the user wants a user profile {
         create user profile in database;
         calculate support values for features using the Apriori
         algorithm based on other similar user profiles;
         store those support values for spatial features in the new
         user profile;
         enable only those features on the map whose support value >
         a specified threshold value;
         IF user logs out {
            record the spatial features enabled by the user;
            check with spatial features enabled in previous sessions
            and store new spatial feature support values in the user
            profile based on the spatial features' frequency;
         }
```

```
   } ELSE use collaborative filtering {
         user requested to choose user category;
         check pattern of spatial features enabled by other users of
         the same user category;
         use the Apriori algorithm to retrieve spatial features of
         probable interest to the user;
         enable only those respective spatial features on the map
         presented;
   }
} ELSE IF a regular user {
   retrieve user profile;
   only display spatial features with support value > a set threshold
   value on the presented map;
   IF user logs out {
         record the spatial features enabled by the user;
         check with spatial features enabled in previous sessions and
         store new spatial feature support values in the user profile
         based on the spatial features' frequency;
   }
}
```

3.6 CONCLUSION

In this thesis, the combination of content-based and collaborative information filtering techniques has been proposed to deliver an efficient and effective model for the personalisation of maps. In the combined map personalisation approach, content-based filtering can be used for regular users whose user profile can be collected based on their interactions with the spatial features, and later analysed to infer the user's current spatial feature preferences. On the other hand, collaborative filtering methods can be utilised for occasional users who do not want to be bothered with creating a user profile, or new users who do not have an existing user profile stored. This information filtering technique makes use of the existing user profiles of similar users, to make an inference for a target user's spatial feature preferences. For collaborative filtering, association rule mining, a well known and researched data mining technique, can be implemented to offer probabilities of interest in various spatial features for a target user.

Association rule mining has been used quite extensively in commercial collaborative recommendation systems. The Apriori algorithm, being the best known and most widely used algorithm to mine association rules, has been proposed to determine the spatial features that are commonly retrieved together by users of the same category. Some of the advantages of the Apriori algorithm include, i) relatively easy to implement compared to other data mining algorithms, ii) algorithm is very fast due to the Apriori property, finding all association rules in linear time in certain cases, and exponential time in others, iii) it can be used to mine association rules in quite large databases. With the combined approach to map personalisation, the implementation of a proven algorithmic solution, such as the Apriori algorithm for association rule mining, helps enable map personalisation for new/occasional users based on the spatial features previously retrieved by like-minded users.

CHAPTER - 4

MOBILE PERSONALISATION SYSTEM

4.1 INTRODUCTION

The combined approach for map personalisation has currently been implemented into a working system utilising a distributed, client-server architecture. This system architecture comprises of, i) a map server, using Oracle Spatial Database Management System, ii) an application server, based on Oracle Application Server, and iii) the system mobile users. The application server is used to deploy the GUI and provide a connection between the geo-spatial data on the map server, and the clients requesting the data. It also deploys the map rendering application, which is utilised by the GUI to process requests from users, and generate a personalised map response displayed on clients' mobile devices. Implementing a distributed, client-server architecture, enables the system to be accessed by multiple users simultaneously.

The personalisation of maps, in our system, is achieved through the combined implementation of the two information filtering techniques of content-based filtering and collaborative filtering. Content-based filtering is used for regular users of the system, whereas collaborative filtering is used for new or occasional users. The GUI for mobile devices has been developed using open and standard system development tools, making it non-proprietary and easily accessible by everyday mobile devices. Java programming language has been used to develop the GUI, and hence offers our system platform independence, due to the Java Platform being installed and readily available on practically all PCs and mobile devices in this day and age. Our system also uses standard network communication protocols, such as WAP and HTTP, to communicate between multiple mobile users and the server. This enables our system to be interoperable with other mobile GIS, and be readily accessed by everyday mobile devices.

The remainder of this chapter is organised as follows. Section 4.2 discusses the design and architecture of the implemented system, along with the methods utilised for communication between the various components within the system architecture. Section 4.3 presents the geo-spatial map datasets used, and the processes executed for the set up of those datasets in our system. Section 4.4 outlines the major tables existing in the spatial database, and their role in the functioning of our system. In Section 4.5, the implementation of a combined approach for map personalisation is discussed. The application server, responsible for the deployment of the GUI for mobile device users, and the map rendering application, is detailed in Section 4.6. Section 4.7 describes the implementation of the system on the client side consisting of mobile device clients as well as desktop PC clients. In Section 4.8, the functionalities of the GUI for mobile devices are discussed, and finally, a conclusion is drawn in Section 4.9.

4.2 SYSTEM DESIGN AND ARCHITECTURE

A few of the currently available mobile map systems, such as Google Maps for Mobile [28], J2ME Map [48], iMapia [29], and Yahoo! Local for mobile [86] provide maps and driving directions from a user's current location to a destination. They provide a comprehensive amount of services and features regarding the manipulation and interaction with stored maps. However, personalising the stored maps in order that only information relevant to each individual user is retrieved is not available. Personalisation exists by storing favourite locations/maps and providing a history of recent locations visited. Our system personalises maps for individual users based on each user's current spatial feature interests or preferences.

A lot of commercially available GIS come pre-installed in proprietary handheld devices such as, TracerTrak® asset tracking solutions, TomTom automobile navigation devices and Trimble Nomad GIS/GPS Data Loggers. Such GIS are normally equipped with fast processors and adequate memory resources. Having maps/geo-spatial services pre-loaded onto a handheld device allows a GIS to operate without connecting to a costly wireless network. However, such GIS typically have quite specific fields of application, such as in-car navigation systems, asset tracking systems, field data loggers, etc. Another disadvantage is the delay on the loading of new map datasets or system updating process, which is often either too costly or time consuming.

Recently, due to major advancements in wireless and mobile networks technology, such as the introduction of 3G/3.5G mobile networks, mobile GIS have been developed based on distributed client server architectures [1, 21, 35, 36, 43, 44, 58]. These architectures comprise of clients requesting relevant information from a remote server. Comprehensive maps/geo-spatial services stored in the server can be maintained and updated frequently. These maps/geo-spatial services can then be sent to the client's mobile device, across the Internet and mobile networks, upon request. Implementing this client-server architecture for mobile GIS helps address the following issues related to proprietary handheld GIS:

- GIS application can be accessed by proprietary handheld devices only
- Specific application purposes not offering variety of usage, and
- Datasets/systems not updated often enough

Our system has therefore, been developed based on a distributed, four-tier client-server architecture. Figure 4.1 gives a brief visual overview of the system architecture. It consists of a database server at the base of the architecture, connected to an application server in the above tier, followed by the WAP Gateway and mobile device users at the other end, connecting to the application server.

The map server makes use of Oracle Spatial Database 10g, which is used to store the vector map datasets in spatial table format, along with other essential information such as the user profiles and GPS locations of users.

The application server utilises Oracle Application Server (Oracle AS) 10g to deploy web applications. Oracle MapViewer [59], the employed map rendering application, has been deployed in Oracle AS to retrieve the requested geo-spatial information stored in the Oracle Spatial Database, and render those results as maps. The Oracle AS is also used to deploy the GUI for mobile device users, programmed in Java, with the use of a Java Integrated Development Environment (IDE) developed by Oracle, called JDeveloper [60]. A connection is made between the database server and application server via the sending and receiving of SQL requests and responses passed through a Java DataBase Connectivity (JDBC) link. This link allows applications deployed on the application server to access the database and retrieve the desired data.

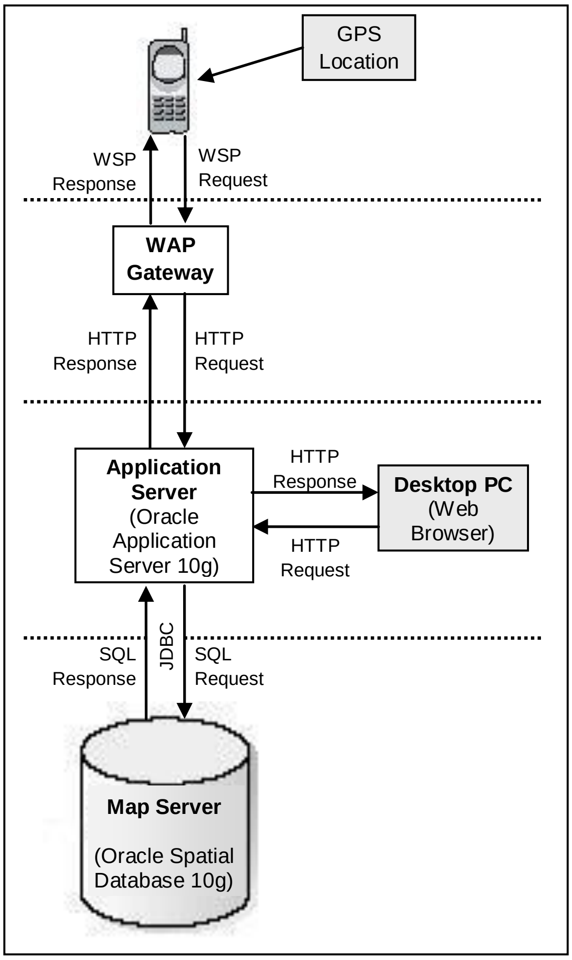


Figure 4.1 - System Architecture

A GUI for desktop PCs was initially developed for use by system developers, providing a wide range of functions. This has been programmed in Java and presented as a platform independent standalone application, which can be installed on practically any desktop PC or laptop. The desktop PC Java application requires a connection to the Internet to retrieve and download the relevant maps/geo-spatial services from the server. Maps/geo-spatial services are retrieved by means of the GUI sending an HTTP request, with the user's current location and other related attributes, to the MapViewer map rendering application deployed on the application server. MapViewer then uses this information to retrieve the required geo-spatial information from the detailed geo-spatial datasets stored in the database, rendering the results as a map. The Uniform Resource Locator (URL) of the map rendered on the server is sent back to the GUI as an HTTP response, and the map is then loaded through accessing the received map URL.

Mobile device users can access the system across mobile networks and the Internet through their mobile devices' respective web browsers. The GUI for mobile device users has been developed by creating JavaServer Pages (JSP) files that are deployed on the application server. JSP files are essentially HyperText Markup Language (HTML) files that allow Java code and certain pre-defined actions to be embedded, providing dynamic content. These JSP files can be accessed through everyday mobile devices' web browsers by connecting to the URL of the GUI deployed on the application server, and then downloading the mobile device GUI. Mobile devices use a WAP gateway to access the Internet. This gateway sits between the mobile devices and the World Wide Web, translating web pages into a format suitable for mobile devices. Therefore, to retrieve a map around the user's current location, the GUI sends a Wireless Session Protocol (WSP) request to the WAP Gateway, for access of information over mobile networks. The WAP Gateway then converts this to an HTTP request, sent to the MapViewer application on the application server. The MapViewer application queries the detailed geo-spatial map dataset, retrieving only the necessary data for the requested map, and renders the resultant map on the server. The URL of this map is sent back as an HTTP response to the WAP Gateway, which then converts and transfers it to the GUI as a WSP response. This URL is used to access the map image and display it on the mobile GUI in a satisfactory manner. Some of the features available in the GUI for mobile devices are briefly outlined below:

- Standard map navigation features, such as zooming, panning, and re-centering
- Enabling/disabling features of interest on the map
- Acquire a user's current location through GPS
- Display a route from user defined location to destination
- Interfaces developed to enable personalisation of maps i.e., login page, creating new user profiles page, etc.

These features are discussed in detail in Section 4.8.

4.3 SYSTEM DATASETS

Our system uses real world vector geo-spatial datasets of a segment of Auckland region, provided by Goldfinger HighTech Ltd. Goldfinger HighTech [27] is a leading developer of world-class GPS technology and software, successfully launching New Zealand's first, fully mapped and portable GPS car navigator. Vector datasets include geometries of points, lines and polygons to represent the geometry of map features. The points in vector data could be used to show landmarks or locations of amenities, whereas the lines are a series of points joined together to display roads, rivers, railway tracks etc. Polygons are shapes bounded by various lines, to represent housing subdivisions, parks/reserves, universities etc. Using vector datasets allows maps to be displayed as vector graphics, which enables zooming in or out of a map without any degradation in map clarity.

The major geo-spatial feature categories contained in the datasets provided are summarised in Table 4.1.

Table 4.1 - Feature Categories

Feature Category	Type of Geometry
Normal Roads	Polylines
Main Roads	Polylines
State Highways	Polylines
Motorways	Polylines
Petrol Stations	Points
Shopping Malls	Points
Police Stations	Points
Post Shops	Points
Ambulances	Points
Airports	Points
Town Names	Points

4.3.1 DATA CONVERSION AND PROCESS

The vector geo-spatial datasets used in our system were originally obtained in **MapInfo Interchange Format**[8]. However, this format of geo-spatial data is not compatible and cannot be directly loaded into the Oracle Spatial Database used in our system. Therefore, this data had to be converted into spatial tables and loaded into the Oracle Spatial Database. This was done using an application called FME Workbench by Safe Software Inc. [74], which is a tool to perform spatial data format conversion. By using the FME Workbench application, the geo-spatial data was converted from MapInfo Interchange Format and loaded into spatial tables in the Oracle Spatial Database.

This process of converting geo-spatial data consisted of selecting the source and destination dataset needed for the conversion. The source data was selected as the MapInfo table files containing the geo-spatial datasets. The attributes, or columns, of the source data tables were specified for conversion. For the destination data, a connection was made to the Oracle Spatial Database specifying the table to be created in the database. Once these parameters were set up correctly, the data conversion process could be commenced, and upon successful completion, the required geo-spatial data was present in the table specified in the Oracle Spatial Database.

Since Oracle Spatial Database is an object-relational database, the spatial vector geometries, such as points, lines and polygons, were stored in their respective tables as spatial objects. Each row in the spatial tables represents a particular spatial object, linked to a real world location, containing geographic latitude and longitude coordinates. This spatial object is stored in a column of object type SDO_GEOMETRY (a spatial object data type). Any table that has a column of type SDO_GEOMETRY must have at least another column, or set of columns,

[8] **MapInfo Interchange Format (MIF)** is a map and database exporting file format of the MapInfo software product. It is one of the most commonly used geographic formats and also well supported by almost every GIS product. MIF is an American Standard Code for Information Interchange (ASCII) text file format that fully describes the contents of a MapInfo table. The MIF physically consists of two files; a .mif file which contains the geometry and a .mid file which contains the attribute data.

that defines a unique primary key for that table. In our spatial tables, the primary key is associated with a column called LINK_ID which contains a unique ID for each spatial object. See Table 4.2.

Table 4.2 - Oracle Spatial Table

	LINK_ID	NAME	HWY	GEOM
716	109277	ONEHUNGA MALL	NORMALRD	MDSYS.SDO_GEOMETRY
717	110745	PRINCES STREET	NORMALRD	MDSYS.SDO_GEOMETRY
718	109159	NEILSON STREET	MAINRD	MDSYS.SDO_GEOMETRY
719	89041	ORPHEUS DRIVE	NORMALRD	MDSYS.SDO_GEOMETRY
720	94869	HUGH WATT DRIVE	MOTORWY	MDSYS.SDO_GEOMETRY
721	94870	HUGH WATT DRIVE	MOTORWY	MDSYS.SDO_GEOMETRY
722	109160	NEILSON STREET	MAINRD	MDSYS.SDO_GEOMETRY
723	110319	HUGH WATT DRIVE	MOTORWY	MDSYS.SDO_GEOMETRY
724	110739	PRINCES STREET	NORMALRD	MDSYS.SDO_GEOMETRY
725	108477	CHURCH STREET	NORMALRD	MDSYS.SDO_GEOMETRY

Oracle Spatial defines the spatial object type SDO_GEOMETRY as:

```
CREATE TYPE sdo_geometry AS OBJECT (
 SDO_GTYPE NUMBER,
 SDO_SRID NUMBER,
 SDO_POINT SDO_POINT_TYPE,
 SDO_ELEM_INFO SDO_ELEM_INFO_ARRAY,
 SDO_ORDINATES SDO_ORDINATE_ARRAY);
```

SDO_GTYPE indicates the type of geometry (e.g. point, line/curve, polygon etc.) and dimensions for the object. SDO_SRID is associated with the coordinate system (spatial reference system) of the particular object. SDO_POINT is used to store the geographic location of a point type geometry. If SDO_ELEM_INFO or SDO_ORDINATES is not null then the SDO_POINT value is ignored. SDO_ELEM_INFO is an array of numbers that informs the system of how to interpret the values stored in the SDO_ORDINATES array. Information such as the location of the coordinates for multiple geometries stored in SDO_ORDINATES, are stored in the SDO_ELEM_INFO array. And finally, SDO_ORDINATES is the array of numbers that make up the coordinates of the boundary of a spatial object.

Storing the geographic locations of features as spatial objects in tables like the one shown in Table 4.3, makes it easy to retrieve and manipulate the data, while performing queries such as “finding all petrol stations within a 3 kilometre radius from my current location”.

Table 4.3 - Spatial Table of Petrol Stations

	NAME	TYPE	GEOM
1	Mobil St Lukes	Petrol Station	MDSYS.SDO_GEOMETRY
2	Mobil Eden Park	Petrol Station	MDSYS.SDO_GEOMETRY
3	Mobil K' Rd	Petrol Station	MDSYS.SDO_GEOMETRY
4	Mobil Quary St	Petrol Station	MDSYS.SDO_GEOMETRY
5	Mobil Parnell	Petrol Station	MDSYS.SDO_GEOMETRY
6	Mobil Onehunga	Petrol Station	MDSYS.SDO_GEOMETRY
7	Mobill Mangere	Petrol Station	MDSYS.SDO_GEOMETRY
8	Mobil Walmsley Road	Petrol Station	MDSYS.SDO_GEOMETRY
9	Mobil Greenlane	Petrol Station	MDSYS.SDO_GEOMETRY
10	Mobil Mt Wellington	Petrol Station	MDSYS.SDO_GEOMETRY
11	Mobil Mt Richmond	Petrol Station	MDSYS.SDO_GEOMETRY
12	Mobil Puhinui	Petrol Station	MDSYS.SDO_GEOMETRY
13	Mobil Panmure	Petrol Station	MDSYS.SDO_GEOMETRY
14	Mobil Glen Innes	Petrol Station	MDSYS.SDO_GEOMETRY
15	Mobil St George	Petrol Station	MDSYS.SDO_GEOMETRY

The purpose of an index is to provide a method that limits general searches. A spatial index has the same purpose as any other index; the only difference is that the method used is based on spatial criteria, such as intersection and containment. Therefore, in order to provide faster retrieval of the geo-spatial datasets, in our system, spatial indexes were created on all geometry tables in the Oracle Spatial database. The spatial indexing method used is known as the R-tree method, which is typically a preferred method for indexing spatial data, and the one used by Oracle Spatial. It uses a Minimum Bounding Rectangle (MBR) to group together objects, such as points, lines and polygons/shapes. All the objects are then added to an MBR within the spatial index, which leads to the least increase in its size.

4.4 SPATIAL DATABASE MANAGEMENT TABLES

This section will detail the various tables used for the implementation of the system, present in the Oracle Spatial database.

4.4.1 ROAD NETWORK GEOMETRY TABLE

This table consists of the spatial geometries, in the form of polylines, which are used to draw the road network layout of the map. Each row in the table consists of the definition of a polyline, such as the ID of the polyline, the road name, the type of road (motorway, main road, footway etc.), and the geometry of that polyline, represented as a spatial object. A particular road may contain more than one polyline depending on the amount of intersections with other roads.

4.4.2 POINTS OF INTEREST GEOMETRY TABLE

A separate spatial table exists for each Point Of Interest (POI) category. POIs consist of all the amenities and landmarks in the geo-spatial dataset that are represented as a point on the map. These include petrol stations, shopping malls, post shops, etc. Each row of these tables represents an individual amenity or landmark. It consists of a unique ID, name of the amenity or landmark, and the spatial object corresponding to the geographic location of the POI.

4.4.3 NODES AND LINKS TABLE FOR ROUTING

In order to provide routing functionality in our system, a Network Data Model (NDM) needed to be constructed for the entire road network in the dataset. After converting the geo-spatial data from MapInfo Interchange Format and loading it into their respective Oracle Spatial database tables, the data comprised of a set of polylines that made up the entire road network. The minimum requirements to create an NDM, however, were two spatial tables, one for the links of the road network, and the other for the nodes at the beginning and end of each link. In order to create an NDM, we had to formulate a table for the nodes of the network, corresponding to the links or polylines available in the present table. Therefore, by constructing a few SQL queries to create a "node" at every intersection of "links", the NDM was able to be produced. A spatial table for all the nodes in the NDM was created, consisting of the node ID and the geographic location of each node. Another separate spatial table for the links was also created. This table contained columns for the link ID, spatial geometry of the link, the beginning node ID for the link, the finishing node ID for the link, and whether the link represents a one-way or bi-directional road (see Table 4.4).

This NDM for the road network is used to find a route from a specified location to a destination, based on the A* search shortest path algorithm first described in 1968 by Hart et al. [61]. Figure 4.2 shows the NDM loaded in an application called NDM Network Editor. It provides a visual representation of the NDM and makes it possible to edit or create nodes and links, as well as perform analyses on the NDM such as, 'shortest path' and 'nearest *n* nodes from a specified node'. Highlighted in Figure 4.2 is one particular link of LINK_ID 98444, and its start and end nodes, with node IDs 9842 and 9831 respectively. The corresponding database entry for the link and the nodes is highlighted in Table 4.4.

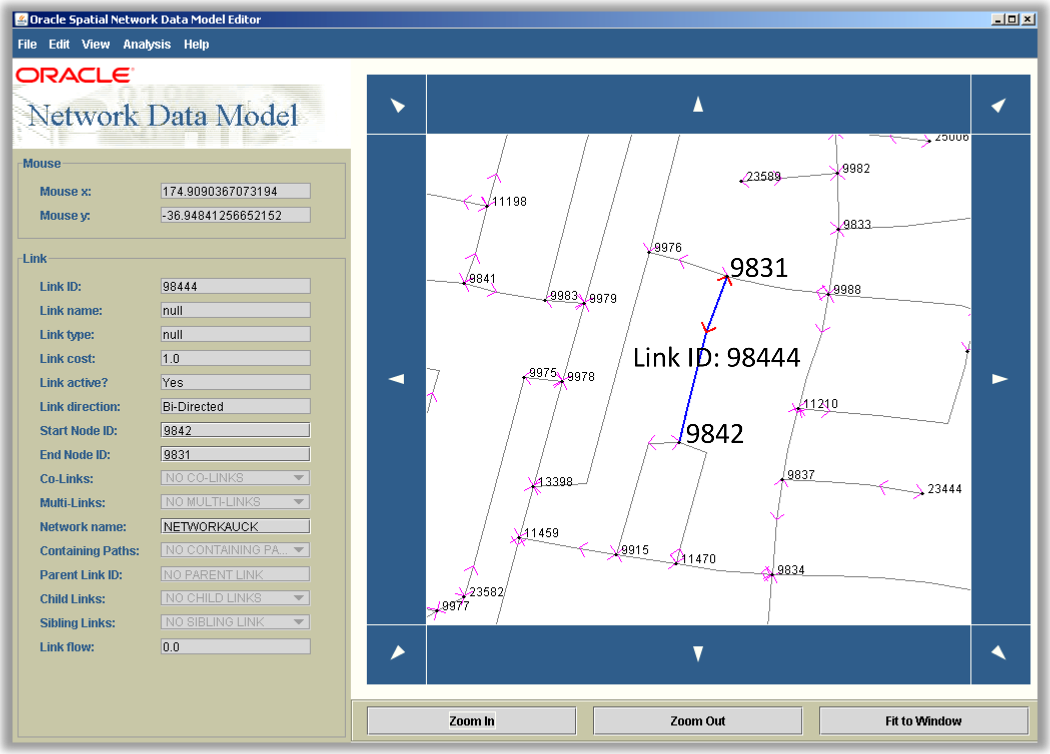


Figure 4.2 - Road NDM Showing Nodes and Links

Table 4.4 - Corresponding Nodes and Link in Links Table

	LINK_ID	GEOMETRY	START_NODE_ID	END_NODE_ID	BIDIRECTED
3326	98438	MDSYS.SDO_GEOMETRY	10476	10404	Y
3327	98439	MDSYS.SDO_GEOMETRY	10404	10398	Y
3328	98440	MDSYS.SDO_GEOMETRY	10398	10401	Y
3329	98441	MDSYS.SDO_GEOMETRY	10401	10560	Y
3330	98443	MDSYS.SDO_GEOMETRY	10486	11461	Y
3331	98444	MDSYS.SDO_GEOMETRY	9842	9831	Y
3332	98445	MDSYS.SDO_GEOMETRY	10490	11461	Y
3333	98446	MDSYS.SDO_GEOMETRY	3088	3148	Y
3334	98447	MDSYS.SDO_GEOMETRY	10332	25255	Y
3335	98448	MDSYS.SDO_GEOMETRY	10270	12512	Y
3336	98449	MDSYS.SDO_GEOMETRY	6790	1249	Y

4.4.4 GPS LOCATIONS TABLE

The GPS Locations table is a simple table designed to store a user's current location acquired through GPS. It consists of columns to store the user's ID, latitude/longitude location, a location number indicating the most recent location, and the time when the location was received. The user's current location is retrieved via a Bluetooth GPS Receiver connected to a developed Java 2 Platform, Micro Edition (J2ME) client application. When a fix has been obtained, the latitude and longitude are parsed and extracted from the stream of GPS sentences, before being sent across mobile networks and the Internet to be inserted into the GPS Locations table. The latitude and longitude values are then retrieved by the GUI for mobile devices and used to display the user's most recent location on the map. This client J2ME application runs in the background on the user's mobile device as they use the GUI. The user's current location is sent across to the server in intervals of about 20 seconds. The reason for this is due to the fact that our system has not been designed solely for the purpose of navigation, thus there is no need for the GUI to update the user's current location more frequently than 20 seconds.

4.4.5 User Profiles Table

In our system, a new user can create and store his/her user profile in the database. These user profiles are stored as rows in the user profile table. The user's unique user ID along with their username, password and user category that he/she belongs to (tourists, residents or surveyors) are stored in this table. This table is also used to store the "support" value (between 0 and 1) associated with each feature type, e.g. shopping malls, streets, motorways, etc. (see Table 4.5). The calculation of this support value is discussed in detail in Section 4.5. As the user uses the system more frequently, the feature support values are updated, giving a clearer indication of the user's preferences regarding spatial feature content.

Table 4.5 - User Profiles Table

USERID	USERNAME	FEATURES	SUPPORT	USERTYPE
5	avibooks	24/7 Petrol Stations	0	Tourist
5	avibooks	Airports	0.4	Tourist
5	avibooks	Ambulances	0	Tourist
5	avibooks	Main Roads	1	Tourist
5	avibooks	Motorways	0.9	Tourist
5	avibooks	Normal Roads / Streets	0.8	Tourist
5	avibooks	Petrol Stations	0.2	Tourist
5	avibooks	Police Stations	0	Tourist
5	avibooks	Post Shops	0.5	Tourist
5	avibooks	Shopping Malls	0.6	Tourist
5	avibooks	State Highways	0.9	Tourist

4.5 PERSONALISATION IMPLEMENTATION ON THE MAP SERVER

In our system, the personalisation of maps is offered for various interests of users; users who frequently use the system and have their user profile stored on the server, users who are using the system for the first time and want to create a user profile for themselves, and users who are infrequent or occasional and do not wish to have their user profile stored. The main purpose of map personalisation is to extract relevant data from a detailed/comprehensive geo-spatial dataset, providing the user with a personalised map/geo-spatial service containing reduced data as compared to the generalised dataset. This helps in various ways; i) it reduces the information overload on mobile devices' small screens and improves the legibility of the maps displayed, ii) it reduces the time and cost to transfer the maps across costly mobile networks and the Internet.

To offer map personalisation in our system, two information filtering techniques have been implemented, namely content-based filtering and collaborative filtering. Content-based filtering is used for regular users of the system who have their user profile stored on the server. Collaborative filtering techniques using an advanced inference technique, known as association rule mining, have been implemented to discover interesting and useful relations between spatial features most commonly retrieved by similar users in the system. In our system, collaborative filtering has been implemented for first time users of the system who want to create a user profile, and for infrequent/occasional users who do not wish to have their user profile stored. The following sections will explain how these information filtering techniques have been implemented for each type of user.

4.5.1 REGULAR USERS WITH A USER PROFILE

Content-based filtering is implemented for users who agree to have their own individual user profile stored in the system database on the map server (discretely located and consisting of geo-spatial datasets and user profiles). Upon creation of a user profile, the system requests very few details from the user, mainly because people would not be interested in spending time filling out a lengthy questionnaire. The system requests the user to input the following details; the user's login information, consisting of their user ID, username and password, and a user category best describing the user (tourist, surveyor or local resident). At present no security measures regarding the transmission or storage of the user's password have been implemented, since our system does not store any personal information about the user. The presence of the user password is merely for determining which user is attempting to log in to the system. Password security measures may be implemented at a later stage if further information about the user is required. This data is then stored into the user profiles table located in the system database. Based on the selected user category, support values (values between 0 and 1) associated with each feature type (such as motorways, petrol stations, post shops etc.), are also calculated and stored in the user profile. The calculation of these initial support values is based on a technique known as associations rule mining, which calculates support values for feature types, based on the support values stored in user profiles of other similar users of the system. Association rule mining is primarily used for collaborative filtering techniques in our system, and is discussed in greater detail in Section 4.5.2. A support value of 1 indicates greatest interest in the feature, whereas a support value of 0 indicates no interest at all in that feature. These support values, associated with each feature type, are constantly updated after every session of usage with the system, based on the user's current and past interactions with the respective spatial features. The next time the user logs in to the system, a personalised map, containing only spatial features with support values greater than a specified threshold value, are displayed.

There exists a separate "Transaction" table in the database (see Table 4.6) that is used to monitor the features present at the end of each user session (between when the user logs in and logs out) with the system. This table is also used to record the latitude and longitude location of the displayed map when each user logs off. This makes it possible to provide the user with a personalised map at the last known location of the user when they log in. Once the user logs off from the system, the spatial features enabled by the user in the current map session are stored in the transaction table, along with the location of the last map frame displayed, a shown in Table 4.6.

Table 4.6 - Transaction Table in Database showing last ten transactions for UserID: 5

TRANSID	USERID	USERTYPE	FEATURES	LATITUDE	LONGITUDE
7	5	Tourist	normal main state motor shop	-36.93826	174.896646
8	5	Tourist	normal main state petrol post airport	-36.91706	174.917046
9	5	Tourist	normal main state motor shop airport	-36.89786	174.906446
10	5	Tourist	normal main state motor post	-36.86866	174.887646
11	5	Tourist	normal main state motor shop airport	-36.96026	174.918446
12	5	Tourist	normal main motor post shop	-36.89786	174.881846
13	5	Tourist	normal main state motor post shop	-36.92206	174.895446
14	5	Tourist	main state motor post airport	-36.92206	174.895446
15	5	Tourist	main state motor petrol shop	-36.92206	174.895446
16	5	Tourist	normal main state motor	-36.95374	174.909074

These transactions are then used to calculate and update the support values for each spatial feature type stored in the respective user's profile. The support values are updated as follows; for a particular user, the frequency of occurrence of each spatial feature is checked in the transaction table. In order to provide support values for spatial features based on the user's current preferences and requirements, if more than ten transactions exist for that user, only the last ten transactions are taken into consideration. This is due to the fact that a user's preferences may change over time, and by using only the last ten transactions, it reduces the possibility of a user's previous spatial feature preferences or interests being taken into account. Based on the frequency of occurrence of a particular spatial feature *f*, its support value S stored in the user profile is calculated and updated as:

Equation 4.1 - Spatial Feature Support Value

$$S = \frac{f}{T}$$

where, *T* is the total number of transactions stored in the transaction table for that user, if less than ten. For instance, referring to Table 4.6, to calculate the support value of the spatial feature "Shopping Malls" for a user with "UserID" 5, the system checks the frequency of the feature "Shopping Malls" (denoted by "shop" in the transactions table) being present in the transactions table. In the transactions table in Table 4.6, the frequency of occurrence for "shop", as we can see, is six times in the last ten transactions. Therefore, the support value for the spatial feature "Shopping Malls" is:

$$S(shop) = \frac{f(shop)}{T} \quad \rightarrow \quad S(shop) = \frac{6}{10} = 0.6$$

This value is reflected in the User Table shown in Table 4.5.

After every user session with the system, this process of updating feature support values, is repeated, providing us with a better perception of a user's possible changing preferences regarding spatial feature content retrieval.

4.5.2 OCCASIONAL USERS WITHOUT A USER PROFILE

To provide personalisation to users who do not wish to store a user profile on the system proved to be quite challenging in the implementation of our system. When the user chooses not to create a user profile or is a one-off user of the system, the user is requested to select which user category best describes them, i.e., tourist, surveyor or local resident. Based on the user's response, the GUI connects to the database and checks the support values of the various features in the user profiles of all users belonging to the same user category. We then calculate average support values for each spatial feature as follows. For each spatial feature *F* present in all user profiles of user category *C*, its average support value F_{avg} is:

Equation 4.2 - Average Support Value

$$F_{avg} = \frac{F_{C1} + F_{C2} + F_{C3} + \ldots + F_{Cn}}{n_C}$$

where, n is the number of user profiles belonging to the user category C, and F_{Cn} is the support value of feature F in the n^{th} user profile belonging to user category C.

Once the average support value of all features within the selected user category are found, the two features containing the highest and second highest average support values can be used as the antecedent of all subsequently generated association rules. At this stage, the Apriori algorithm for association rule mining is activated to find the probability of every other spatial feature being retrieved, provided the two spatial features with the highest average support value are present. The Apriori algorithm is an iterative algorithm and is broken down into two main steps; i) to find all the frequent itemsets that are retrieved together, provided they meet the minimum support threshold, and ii) use those frequent itemsets to generate association rules.

A Java implementation of the Apriori algorithm has been used in our system. The algorithm uses the transaction data stored in the Transactions table, as shown in Table 4.6, to calculate support values for the remainder of the spatial features. Based on the user category selected by the user, the system checks all the transactions from all users belonging to that same user category. This data is then used as the transaction database for the Apriori algorithm to find association rules for the spatial features commonly retrieved together.

A threshold minimum support value of 50% is currently set to filter out only the features retrieved together frequently. Along with the two previous themes, all other themes consisting of a support value/probability greater than 50% and a confidence value greater than 60% are enabled and displayed on the personalised map.

4.5.3 NEW USERS

In order to offer map personalisation to new users of the system who have just created their user profile, we again incorporate the Apriori algorithm of association rule mining.

When a new user wishes to store his/her user profile on the server, the system stores their login information in a newly created user profile; along with the user category they choose (tourist, surveyor or local resident). For each spatial feature in the new user's profile, a support value needs to be calculated. Similar to the collaborative filtering technique in the previous section, Section 4.5.2, we need to find two spatial features with the greatest average support value. This can be done by using Equation 4.2 and following the steps outlined in Section 4.5.2. After finding the two spatial features with the highest average support values, these features, along with their support values, are inserted into the new user's profile.

To calculate the rest of the feature support values, the Apriori algorithm is used. By employing the algorithm against the transactional database of the new user's category, with a minimum support value of 0.5, it is possible to extract all relevant association rules along with the support and confidence values of each rule. Only the necessary association rules need to be extracted from the resultant rules. Association rules are only required in the following format:

Equation 4.3 - Association Rule Format

$$\{F(Highest\ s_{avg})\ AND\ F(2ndHighest\ s_{avg})\} \Rightarrow \{F_n\}$$

where, *F(Highest* s_{avg}*)* is the feature with the highest support value for the specified user category, and *F(2ndHighest* s_{avg}*)* is the feature with the second highest support value for the specified user category. F_n is each feature not included as the feature with the highest or second highest average support value. The support value for each association rule is regarded as the support value for that rule's spatial feature F_n, and is respectively inserted into the new user's profile. The spatial features not included in F_n represent spatial features with support values less than 0.5, hence being considered irrelevant, and a support value of 0 is assigned for those features in the user profile.

After the completion of this process, a personalised map is generated for the new user, displaying only features containing support values greater than a specified threshold value.

4.6 ORACLE APPLICATION SERVER

The application server in the system utilises Oracle AS 10g. Oracle AS 10g provides a means for developing, integrating and deploying enterprise applications, portals, and Web services. The application server in our system was used to establish a connection between the information stored on the database, and the clients. Via a JDBC link between the application server and the database server, communication between these servers was made possible. Since JDBC is a widely used standard communication link, this allows interoperability of our system with other mobile GIS, enabling our GUI to access the databases of other GIS and vice versa. An application called Oracle Application Server MapViewer (Oracle AS MapViewer) was chosen as the map rendering application, deployed on the application server. Upon a map request from mobile device or desktop PC users, Oracle AS MapViewer processes the request, queries the corresponding geo-spatial data from the database, renders the result as a map and sends it back to the client as a map response. The application server is also used to deploy the GUI developed for mobile devices, which makes it possible for mobile device users to access the GUI through the Internet and mobile networks, anytime and anywhere.

4.6.1 ORACLE APPLICATION SERVER MAPVIEWER

MapViewer is a service for rendering maps making use of the geo-spatial data stored in an Oracle Spatial Database. Oracle AS MapViewer provides a comprehensive set of functions and tools that help to minimise the complexity of spatial data queries and at the same time simplifies cartographic rendering [59]. While hiding these complexities, it also provides highly customisable options and features for the more advanced and experienced users. Oracle AS MapViewer is written purely in Java and runs in a Java 2 Platform, Enterprise Edition (J2EE) environment.

In our system, Oracle AS MapViewer is used to retrieve the spatial vector map data stored in the Oracle Spatial Database, and render a map based on the preferences and current location of the user. Oracle AS MapViewer provides a complete set of Application Programming Interfaces (APIs) that are used to manipulate the geo-spatial data according to users' requests. These Java APIs are used in the development of our GUI, to provide maps/geo-spatial services to users and to perform various functions within the GUI, such as, i) panning and re-centering the map, ii) zooming in/out of the map, iii) enabling/disabling of particular spatial feature categories on the map, iv) setting the attributes regarding the display of the map, i.e., width, height, center, size, scale etc., and v) providing routing from location to destination. When a user requests for a particular map area, this is converted into an XML map request containing all the necessary attributes, such as the title, size, and center of the map. Oracle AS MapViewer uses this information to retrieve the relevant geo-spatial data from the database, renders the resultant map and stores it on the server. Figure 4.3 shows a map request in XML sent through the Oracle AS MapViewer web interface and Figure 4.4 displays the resulting map of that XML request.

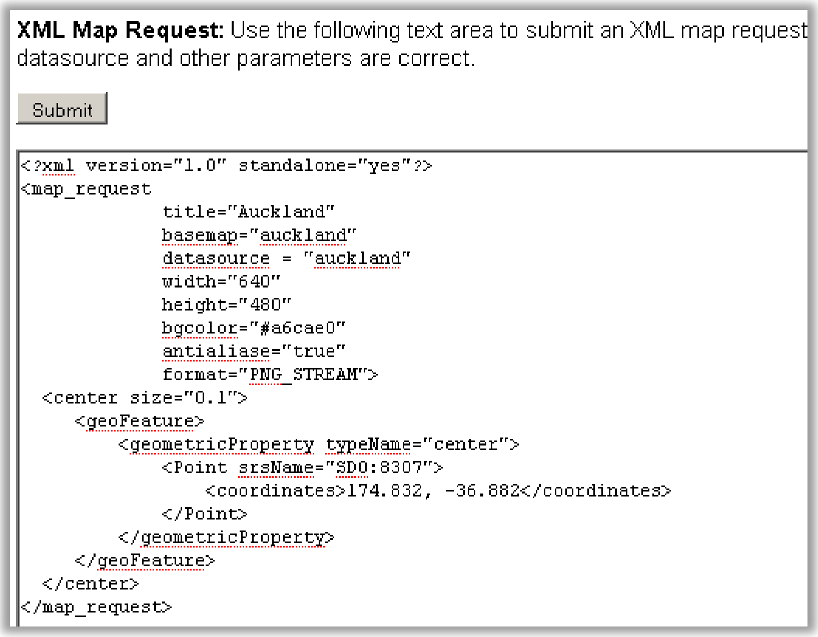


Figure 4.3 - XML Map Request in MapViewer

Figure 4.4 - Corresponding Map Response

Oracle AS MapViewer also includes a Java application tool called Oracle MapBuilder, which can be used to dynamically create the styles and themes necessary to build maps that are displayed to the users. Its primary function is enabling users to visually create and manage mapping metadata. It can be used to create XML style definitions, or modify definitions of pre-existing styles, categorised into colour, line, marker, text, or advanced styles. The desired styles are visually created by interacting with the interface provided in the program. These styles are attributed to certain queried results from the database, forming geometric themes that can be previewed in a map. A base map can then be formed by a collaboration of selected themes. All the map metadata created in MapBuilder is stored in the Oracle Spatial database as XML definitions.

4.7 SYSTEM IMPLEMENTATION ON THE CLIENT SIDE

This system is based on a distributed four-tier, client-server architecture as discussed previously. In this system, GUIs were developed for mobile devices and desktop PCs. The GUI for mobile devices was deployed on the application server, which allowed users to access it on their mobile devices' web browsers through mobile networks and the Internet. The GUI for desktop PCs was developed as a standalone Java application that can be installed and used on any PC, to download the maps/geo-spatial services from the server via the Internet. The GUI for mobile devices has been programmed in Java and HTML and can be accessed by any mobile device consisting of a mobile web browser. The GUIs can be accessed by entering their respective URLs into the web browser, which then retrieves and displays the content of the respective GUI from the application server.

4.8 GRAPHICAL USER INTERFACES AND SYSTEM FUNCTIONS

Easy to use GUIs have been programmed in Java for everyday mobile device users as well as desktop PC users. The GUI for mobile devices has been created as a JSP file, which provides Java programming capability inside an HTML page. It is deployed on the application server, and accessed through mobile networks and the Internet, by entering the URL in the mobile device's built in web browser. The GUI for desktop

PC users is created as a standalone Java application that can be run from any desktop PC or laptop. Upon a particular map request made by a user, it is sent across mobile networks and the Internet, to the application server. The map rendering application processes the received map request and retrieves the relevant data, based on the user's preferences stored in their profile, from the database. This data is then rendered as a map and sent back to the client as a map response. The GUI for mobile devices has been specifically designed taking into account mobile devices' small screens, limited memory capabilities and restricted input methods, as shown below in Figure 4.5.

Figure 4.5 - Personalised Map Displayed on Nokia N81 and LG KF750

Some of the functions developed for the mobile device GUI, which are also present in the desktop PC GUI are outlined below.

4.8.1 INITIAL PAGE FACILITATING MAP PERSONALISATION

The interfaces developed to facilitate the personalisation of maps are some of the key features of our system. The initial page loaded in the GUI provides the user with various options to access the system (Figure 4.6). Regular users of the system can login by entering their User ID and password (provided at the account creation stage). A personalised map based on the preferences in their user profile will then be retrieved and displayed, while their user profile is subsequently updated when they log off from the system.

New/occasional users are provided with the option of either storing a user profile for themselves on the system, or choosing to access the system without creating a user profile. If the user does wish to create a user profile, simple login details such as the user ID, username and password, along with the user category (tourist, surveyor or local resident) they best belong to, are requested and stored in the database. After creation of the user profile, a personalised map, based on the spatial feature preferences of other users belonging to the user category specified, is retrieved. However, if the user does not wish to create a user profile, the personalisation of maps is achieved by retrieving the spatial features most commonly retrieved together by other similar users in the system, using a well known data mining technique known as association rule mining.

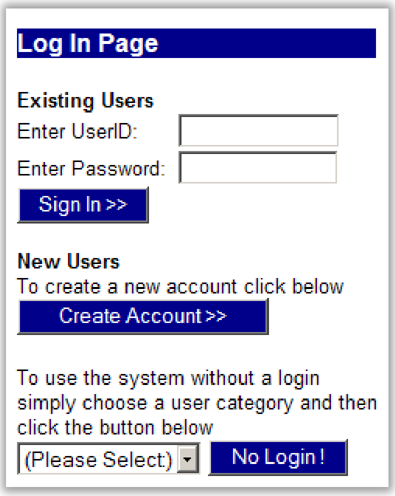


Figure 4.6 - Login Page of Mobile Device GUI

4.8.2 ENABLING / DISABLING FEATURES OF INTEREST

Users have the option of directly enabling/disabling any of the features provided within the map interface, such as motorways, petrol stations, city streets, shopping malls etc. Providing users with the ability to turn on/off particular spatial features of interest is crucial in the inference of a user's preferences regarding spatial feature content retrieval. By recording and analysing the spatial features selected by a particular user, we are able to update the support values for the respective features stored in their user profile accordingly. This data can also be used to retrieve spatial features for users who do not wish to store their user profile by determining the spatial features most commonly accessed by a particular group of users. Figure 4.7 shows the GUI displaying only the spatial features presented to a particular user of the system. The features enabled on the map are highlighted in yellow, which are normal roads, main roads, motorways, petrol stations, 24/7 petrol stations and post shops.

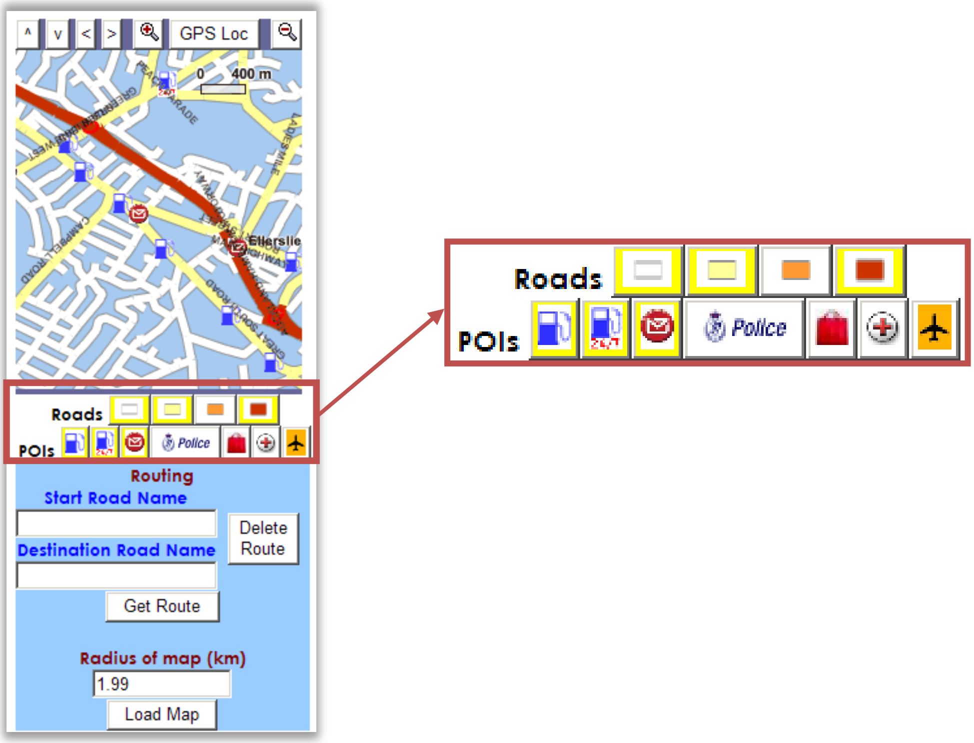


Figure 4.7 - Features Enabled for Display on Map

4.8.3 USER'S CURRENT LOCATION THROUGH GPS

To get the user's current location, a J2ME application has been developed, which connects to a Bluetooth GPS Receiver and stores the user's location information into a table in the database. This information can then be retrieved by the GUI and displayed on the map (Figure 4.8). To enable the J2ME application to connect to the Oracle Spatial Database and write the user's current location into the table, a Java Servlet needed to be formulated. A servlet is a Java class that processes requests and generates responses based on that request. Servlets, unlike Java applets[9] run on the server side instead of the client. The purpose of the developed servlet was to receive the latitude and longitude location information from the J2ME application and then connect to the Oracle Spatial Database and insert that information into the table. This servlet was also deployed on the application server. The reason for using a servlet to connect to the Oracle Spatial database was due to the fact that the API for J2ME does not support the Oracle Database drivers required to make a connection to the database. Therefore, no direct link between the client J2ME application and the Oracle Spatial Database server can be created without going through a servlet. This J2ME application to retrieve a user's current location was created independent to the web GUI, as it did not seem possible to connect to either an inbuilt GPS receiver or an external one through the web GUI, since the JSP GUI runs on the server.

Figure 4.8 - GPS Location of User Marked on Map

[9] A Java applet is a special kind of Java program designed to be run in a Java-enabled web browser. Applets bring dynamic interaction and live animation into an otherwise static HTML page [82].

4.8.4 ROUTING FUNCTIONALITY

Another vital feature provided in the GUI for mobile devices is the ability to plot a route from any specified location to a destination, within the bounds of the available geo-spatial map dataset. The user is required to indicate the start location road name, as well as the destination road name. The system then verifies the road names with their corresponding spatial objects stored in the database, and plots a route, based on the shortest/least-cost path calculated using the A* search algorithm [61], from the start location to the destination. The user is also allowed to center the map at a particular specified road by simply inputting the name of the road in the GUI. The result is a zoomed in map with the requested road at its center. Figure 4.9 shows two snapshots of the GUI displaying two routes with different start locations and destinations.

Figure 4.9 - Routes Calculated from Location to Destination

4.8.5 STANDARD MAP NAVIGATION FEATURES

The developed GUI also provides standard map navigation features, such as zooming in and out of the map, and panning the map in four directions (north, south, east and west). For mobile device web browsers that present a mouse pointer that can be shifted around with the navigation keys, re-centering the map at any clicked location, is also possible. This can be used alongside the standard map panning features. Besides the standard zooming in/out functions present, personalised maps can also be displayed within a specified kilometre radius around the centre of the map. This provides an alternative method to standard zooming, wherein a user can comprehend the representation of the map in actual physical distance.

4.9 CONCLUSION

The map personalisation approach discussed earlier has been incorporated into a working system, using open and standard system development tools. This system has been structured based on a four-tier, distributed client-server architecture.

Personalisation of maps, in our system, is offered to regular users through content-based filtering, and to new or occasional users through collaborative filtering techniques. Maps personalised through content-based filtering are offered by recording the user's spatial feature preferences at the end of every session with the system. Based on the data collected, support values for each spatial feature are computed and stored in their user profile. On the other hand, personalising maps through collaborative filtering involves the calculated retrieval of particular spatial features for a target user, based on the spatial feature preferences of other similar existing users in the system.

GUIs have been developed for mobile device users as well as desktop PC users. These GUIs have been programmed in Java to offer platform independency, and utilise standard communication protocols such as WAP and HTTP to communicate with the application and map servers, making the system interoperable with other mobile GIS. Running Java and standard communication protocols also makes it possible for everyday mobile devices to access the system, anywhere and anytime.

The GUI for mobile devices has been developed using JSP technology. The functions offered in the GUI developed for mobile devices are as follows; i) provision of standard map navigation features, such as zooming, panning, etc. ii) allowing the enabling/disabling of spatial features, as desired by the user, iii) plotting a route on the map, from a specified start point to a particular destination, based on the shortest path between the two, iv) retrieving the user's current geographical location through GPS, and v) the development of the interfaces for personalisation, allowing regular users to log in to the system, and new/occasional users to either create a user profile, or use the system without creating a user profile.

CHAPTER - 5

SYSTEM EVALUATION

5.1 INTRODUCTION

In this thesis, we have successfully developed a working system based on a distributed client-server architecture, wherein everyday mobile device users are able to access the system through mobile networks and the Internet. Our combined map personalisation algorithm has been integrated into this system to deliver personalised maps to users based on their current location, preferences and requirements. Our system aims to address some of the common issues currently existing with mobile GIS, such as, i) large amounts of geo-spatial data transferred across mobile networks resulting in excessive download times costs, ii) information overload on the small screens of mobile devices, and iii) mobile GIS being typically available for PDAs and smart phones only, rather than everyday mobile devices.

The system evaluation has been carried out in terms of data reduction, processing time and the relevance of spatial feature retrieval to users. A group of eleven users with varied backgrounds, competencies and age were selected for the system evaluation, in order to acquire unbiased results. Each user was requested to perform several transactions (sessions) with the system to obtain a better understanding of their spatial feature preferences, which were collected and stored in their user profiles. The personalised maps provided, and the user profile data collected for those users, have been analysed and discussed in this chapter. The evaluation of our system has overall revealed promising and satisfying results.

The remainder of this chapter comprises of the following structure. Section 5.2 gives an overview of the mobile devices used for our system evaluation. Following which, in Section 5.3, a detailed evaluation of our system in terms of the reduction in data sent to the mobile device user, is presented. Section 5.4 provides an evaluation of our system in terms of the time taken to access the system and transfer data between client and server. In Section 5.5, the combined map personalisation algorithm developed is evaluated by a group of users, the results of which are detailed. Section 5.6 provides a discussion on the results of a system performance survey conducted by the users evaluating the system. Section 5.7 discusses the possibility of the interoperability of our system with other mobile GIS. And finally, a conclusion is drawn in Section 5.8.

5.2 MOBILE DEVICES

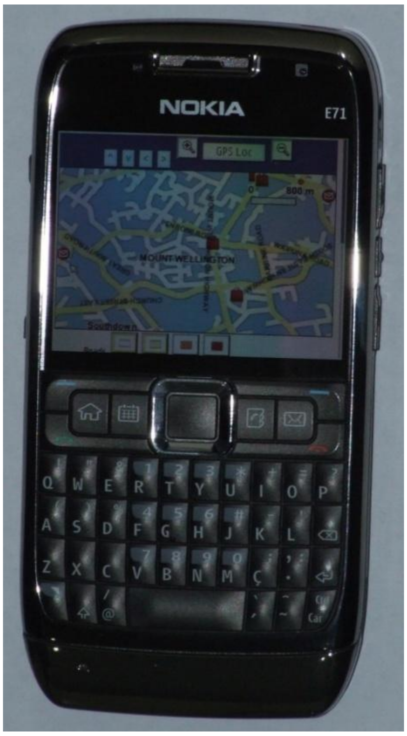
Our system has been successfully tested on a wide range of everyday mobile devices. The main requirement is that, i) the mobile device has a web browser, through which access to the Internet can be available, and ii) the mobile device is Java-enabled, due to the GPS location updating application being developed in Java. These requirements, however, are very common in most mobile devices currently available in today's market, and even average low cost mobile devices would be able to meet these requirements. Due to our system being implemented using open and standard system development tools, and being platform independent, it can be accessed by any Java-enabled mobile device with a web browser.

The mobile devices used for evaluating the system were the Nokia N81, Nokia E71, LG KF750, Nokia 6500 and the Nokia 6121. The configuration of these mobile devices ranges from low level to high level configuration. Our system has the capability to run efficiently on the majority of average everyday mobile devices available at present. The configuration and cost factor of the mobile devices used for the system evaluation has been outlined in Table 5.1. As shown in the table, the cost of these mobile devices ranges from NZ$ 300 to NZ$ 600. However, certain cheaper mobile devices will also be able to access our system, depending on the specifications of the web browser provided with the phone.

Table 5.1 - Configuration and Cost of System Mobile Devices

	Mobile Devices				
	 Nokia N81	 Nokia E71	 LG KF750	 Nokia 6500	 Nokia 6121
Configuration	Mid Level	High Level	Mid Level	Low Level	Low Level
Cost (NZ$)	470.00	570.00	350.00	400.00	300.00

A snapshot of our developed GUI for mobile devices being accessed through the above discussed mobile devices is shown in Figure 5.1. The GUI developed for mobile devices has been tested to be fully functional and operational on all the mobile devices in Table 5.1.

Nokia N81	Nokia E71	LG KF750
		

Nokia 6121	Nokia 6500
	

Figure 5.1 - GUI displayed on Selected Everyday Mobile Devices

The speed of data transfer on the different mobile devices can vary due to recent advancements in mobile network technology. Mobile networks have been constantly evolving, providing mobile users with faster and more convenient Internet access through their mobile devices. Some of the common mobile networks present today include 3G (2000-2005) and 3.5G (2005-present). With the emergence of 3G mobile networks, mobile data transfer speeds were boosted up from their predecessors to average fixed broadband data transfer speeds. Some of the well-known major technologies that form the 3G mobile network are Universal Mobile Telecommunications System (UMTS) and Code Division Multiple Access 2000 Evolution-Data Optimized Revision A (CDMA2000 EVDO-Rev A), that offer data transfer speeds beginning at 144 kbps and reaching peak speeds of 2 megabits per second (mbps). Evolving on the 3G mobile network technologies, the 3.5G mobile network was introduced, providing even better data transfer speeds than their predecessors. The technologies comprised in this network are High-Speed Downlink Packet Access (HSDPA), which evolved from UMTS, and CDMA2000 EVDO-Rev B evolving from CDMA2000 EVDO-Rev A. The data transfer speeds offered by this mobile network ranges from 384 kbps reaching up to 14.4 mbps.

The mobile devices used in the evaluation of our system fall into the various mobile networks previously discussed. Table 5.2 indicates each mobile device used in our system evaluation and their corresponding mobile network technology, providing coverage for that network is available where the GUI is being accessed from. If not the mobile device keeps dropping to a lower data transfer speed mobile network.

Table 5.2 - Mobile Network of System Mobile Devices

Mobile Device	Mobile Network (if coverage available)	Maximum Data Transfer Speed
Nokia 6121	3.5G HSDPA	3.6 mbps / 450 KBps
Nokia 6500	3G UMTS	384 kbps / 48 KBps
Nokia N81	3G UMTS	384 kbps / 48 KBps
LG KF750	3.5G HSDPA	3.6 mbps / 450 KBps
Nokia E71	3.5G HSDPA	3.6 mbps / 450 KBps

Therefore, some newer mobile device models will be able to access and operate our system at greater speeds than their predecessors. However, due to the implementation of map personalisation in our system, wherein only relevant geo-spatial content is retrieved, the time taken to load maps/geo-spatial content is reduced on mobile devices, mainly benefitting everyday mobile devices with slower data transfer speeds.

The methods of input available in mobile devices today vary according to the mobile device's assumed field of usage. Our system GUI only requires a basic mobile keypad for its usage. Most smart phones and PDAs come equipped with a QWERTY[10] keyboard for easier data inputting and convenient selection of functions (Nokia E71, Palm Treo 750, Blackberry Curve 8300 etc.). Some consist of a touch screen, wherein the user is allowed to interact with the interface through a pen/stylus as needed (Apple iPhone, Samsung D980, LG KU990 Viewty etc.). Also, speech instruction recognition has been recently devised as an input method for specific smart phones or PDAs, such as the Apple iPhone 3G S. Developing a GUI for mobile devices, enabling user interaction through these input methods, would offer greater convenience for users of smart phones or PDAs. However, most everyday mobile devices do not come equipped with such high end input method technologies, instead they consist of a simple keypad with numbers from 0-9, and the * and # keys. Therefore, due to mobile devices' varied input restrictions, our GUI utilises standard input methods, which enables all mobile devices to operate the system with no trouble. PDA and smart phone users whose web browsers offer the functionality of mouse pointers or touch screen capabilities will still be able to interact with the GUI through these input methods.

5.3 EVALUATION IN TERMS OF DATA REDUCTION

Personalising maps based on a user's current location, preferences and requirements would result in significant reduction in the amount of geo-spatial data transmitted across mobile networks and the Internet. This, in turn reduces the time and cost of geo-spatial data transmissions across costly mobile networks, as well as prevents

[10] QWERTY is a keyboard layout used in all desktop PCs, laptops and typewriters since the early 1870s. It refers to the arrangement of the keys, and is derived from the first six characters in the uppermost alphabetical row in the keyboard layout.

information overload on the small screens of everyday mobile devices, improving the legibility of the retrieved maps.

Below are two screenshots of maps delivered to a mobile device user at their current location. Figure 5.2 is a fully detailed map with all spatial features enabled, due to and no map personalisation being utilised. Figure 5.3 displays a personalised map consisting of only spatial features that were deemed to be of relevance to the user, based on the spatial feature preferences stored in their user profile. The user profile used to display the personalised map is presented alongside the map in Figure 5.3.

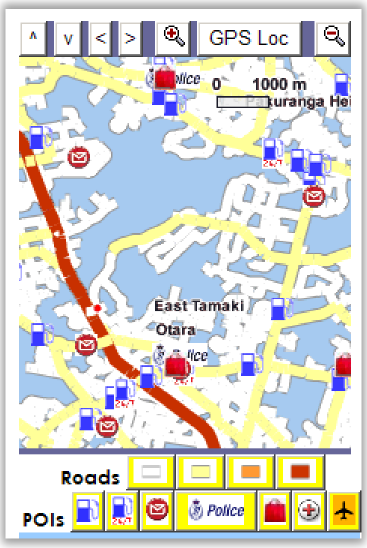
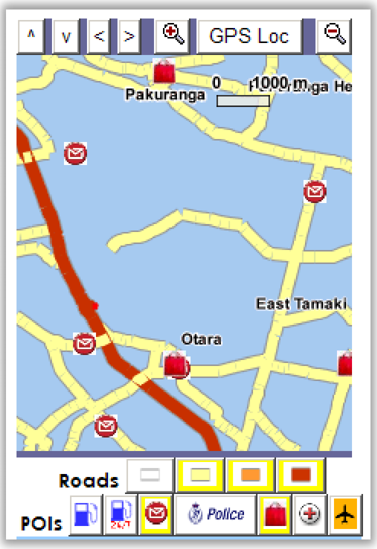


Figure 5.2 - Fully Detailed Map



USERID	USERNAME	FEATURES	SUPPORT
5	avibooks	State Highways	0.7
5	avibooks	Motorways	0.7
5	avibooks	Ambulances	0.55
5	avibooks	Petrol Stations	0.55
5	avibooks	24/7 Petrol Stations	0.3
5	avibooks	Airports	0.5
5	avibooks	Post Shops	0.8
5	avibooks	Shopping Malls	0.75
5	avibooks	Police Stations	0.4
5	avibooks	Normal Roads / Streets	0.4
5	avibooks	Main Roads	0.8

Figure 5.3 - Personalised Map and Corresponding User Profile

As visible in Figure 5.3, the personalised map contains a lot fewer features (main roads, motorways, state highways, shopping malls and post shops) than the fully detailed map displayed in Figure 5.2. The spatial features enabled on the map have a yellow highlight around their representative image beneath the map displayed.

The threshold minimum support value specified to determine the importance of spatial features to the user in Figure 5.3. This value was set at 0.7, which can be altered depending on the desired level of map personalisation. Table 5.3 below gives us an indication of the reduction in data between the fully detailed map and the personalised map.

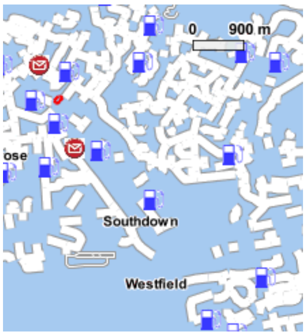
Table 5.3 - Comparing Detailed and Personalised Maps

Performance	Detailed Map (Fig 5.1)	Personalised Map (Fig 5.2)
Data Reduction %	0%	39%
Data Size	20.6 KB	12.6 KB

Based on a randomly selected sample of five personalised maps acquired through the evaluation of our system, we can calculate the average data reduction for a typical personalised map. The data in Table 5.4 analyses the size of each personalised map, compared to the size of the same map containing all spatial features (the fully detailed map).

Table 5.4 - Data Reduction in a Sample of Personalised Maps

<table>
<tr><th>Personalised Map (Map A)</th><th>Fully Detailed Map (Map B)</th><th>Map A Size</th><th>Map B Size</th><th>Data Reduction %</th></tr>
<tr><td>
</td><td>
</td><td>21.7 KB</td><td>28.4 KB</td><td>23.6 %</td></tr>
<tr><td>No. of Feature Categories = 3</td><td>No. of Feature Categories = 7</td><td colspan="3">Total Feature Categories Reduced = 4</td></tr>
</table>

 No. of Feature Categories = 4	 No. of Feature Categories = 9	13.5 KB	18.8 KB	28.2 %
		Total Feature Categories Reduced = 5		
 No. of Feature Categories = 2	 No. of Feature Categories = 9	19.0 KB	26.4 KB	28.0 %
		Total Feature Categories Reduced = 7		
 No. of Feature Categories = 4	 No. of Feature Categories = 8	18.6 KB	27.2 KB	31.6 %
		Total Feature Categories Reduced = 4		
 No. of Feature Categories = 3	 No. of Feature Categories = 8	16.2 KB	24.1 KB	32.8 %
		Total Feature Categories Reduced = 5		

The average reduction in geo-spatial data sent across to users through mobile networks and the Internet, can be ascertained from the information present in Table 5.4. Based on the sample of five personalised maps, the data reduction percentages of each of the personalised maps were 23.6%, 28.2%, 28.0%, 31.6%, and 32.8% respectively. Therefore, the average data reduction percentage is:

$$DR_{avg}(\%) = \frac{23.6 + 28.2 + 28.0 + 31.6 + 32.8}{5} \quad \Longrightarrow \quad DR_{avg}(\%) = 28.8\ \%$$

And the average size of a personalised map is:

$$Size_{avg}(KB) = \frac{21.7 + 13.5 + 19.0 + 18.6 + 16.2}{5}$$

$$Size_{avg}(KB) = 17.8\ KB$$

The data reduction in terms of the number of spatial feature categories retrieved in a personalised map is also quite crucial in reducing the occurrence of information overload on the small screens of mobile devices. In a typical personalised map presented to the user, our system provides an average of about 60% reduction in the number of spatial feature categories displayed. This value is based on the sample of five personalised maps shown in Table 5.4, and has been calculated as follows:

$$FR_{avg}(\%) = \left(\frac{R_{M1} + R_{M2} + R_{M3} + R_{M4} + R_{M5}}{5}\right) \times 100$$

$$FR_{avg}(\%) = \left(\frac{{}^{4}/_{7} + {}^{5}/_{9} + {}^{7}/_{9} + {}^{4}/_{8} + {}^{5}/_{8}}{5}\right) \times 100$$

$$FR_{avg}(\%) = \frac{3.03}{5} \times 100$$

$$FR_{avg}(\%) = 60.6\%$$

where, $FR_{avg}(\%)$ is the average percentage reduction in the number of spatial feature categories retrieved, based on the sample of the five personalised maps shown in Table 5.4, and R_{Mn} is the percentage reduction in the number of spatial feature categories in map *n*.

5.4 EVALUATION IN TERMS OF PROCESSING TIME

Theoretically, the time taken requested to transfer a typical personalised map of size 17.8 KB (based on data in Table 5.4) and the other corresponding images of the GUI (approx. 5 KB) from the server to a user's mobile device using the UMTS mobile network technology is:

$$T(s) = \frac{Size(KB)}{Speed(KB/s)} \quad \Rightarrow \quad T = \frac{17.8\,KB + 5\,KB}{48\,KB/s} \quad \Rightarrow \quad T = 0.48\,seconds$$

This time will be reduced for other mobile devices using faster mobile network technologies. This time of 0.48 seconds for the UMTS mobile network technology, however, is based on the maximum speed allowed on the network, which is most likely to be a lot higher than the average speed at which data is transferred. The average time taken to create a connection to the server and send a map request from the user's mobile device to the server has been recorded at 1.50 seconds. This value has been obtained by checking the time taken to send a user's map request from a mobile device to the server, and taking an average of ten such map requests. The data size of each map request is quite small, however, it takes a reasonable amount of time to get through the WAP gateway and create a connection to the server prior to sending the map request.

The time taken to process each map request on the server, and generate a map to be displayed on the user's mobile device, has been checked and recorded in Table 5.5. This data has been extracted from the application server after a particular system evaluation test carried out by a few users, and indicates the performance of the GUI.

Table 5.5 - Request Processing Performance of the GUI

	GUI Name	
	mainMapInterface.jsp	**createUserProfile.jsp**
Application	webapp1	webapp1
Current Client Processing Time (seconds)	1.96	7.96
Average Client Processing Time (seconds)	2.17	8.16

The GUI indicated by the name mainMapInterface.jsp from Table 5.5 is the main user interface displaying the map and all the mapping functions available in our system. The GUI createUserProfile.jsp represents the page providing the interface to create new user profile. As we can see from the data in Table 5.5, the average time taken for the system to process a request from the client on the main mapping interface (mainMapInterface.jsp) is 2.17 seconds, which is quite satisfactory for client-server mobile GIS.

The total time taken to process a typical user map request is approximately the sum of the time taken to send the map request to the server, the time taken to process the map request on the server, and the time take for the map response to be loaded on the GUI at the client side. This equates to:

$$T_{total} \cong T_{send} + T_{process} + T_{receive} \quad \Longrightarrow \quad T_{total} \cong 1.50 + 2.17 + 0.48$$

$$T_{total} \cong 4.15\ seconds$$

where, T_{total} is the total time taken to process a typical map request, T_{send} is the time taken to send a typical map request to the server, $T_{process}$ is the time taken to process a typical map request on the server, and $T_{receive}$ is the time taken for the map response to be sent and displayed on the client side GUI.

5.5 EVALUATION OF THE COMBINED MAP PERSONALISATION ALGORITHM

In this section, the combined approach to map personalisation, implementing content-based information filtering for regular users and collaborative information filtering for new/occasional users, will be tested and evaluated by eleven individual users. Users with varied skills and age were selected to evaluate the system, in order to provide unbiased results. The aim of this evaluation is to determine whether the spatial features retrieved for regular users, as well as occasional users, match their current spatial feature preferences.

To perform this evaluation, each user was given a set of instructions regarding the correct usage of the system. Along with these instructions, they were each given a brief scenario, wherein the user was allocated a user category (tourist, surveyor or local resident), and to operate the system assuming they belonged to the user category assigned to them. Each user was also requested to access and operate the system at least three to four times in order to get sufficient information about the user's spatial feature interests or preferences. Subsequently, they were asked to fill out a system performance survey regarding their usage of the system. The results obtained from this evaluation are detailed below.

5.5.1 EVALUATION FOR REGULAR USERS

Personalised maps were delivered to the users of the evaluation through content-based filtering, wherein their user profiles were stored on the server and updated with their spatial feature preferences on a session-by-session basis. Once each user had finished evaluating the system, they were presented with a brief system performance survey where they were asked to rate the performance of a few aspects of the system on a scale from 1 to 7, 1 being 'Extremely Poor' and 7 being 'Excellent'. Some of those aspects rated were the easy usability of the map interface, and whether the spatial features retrieved for them, after a few sessions of usage, were relevant to their current spatial feature preferences.

From the results of the system performance survey, we can observe that 55% of the users rated the easy usability of the GUI as 'Excellent', whereas the rest 45% thought it was 'Very Good'. In regards with the spatial features retrieved matching the user's preferences based on current and past usage, 64% of users rated the system as being 'Excellent' in retrieving spatial features based on their preferences, whereas 36% rated the system as being 'Very Good' in that matter. The detailed results of the system performance survey are provided in Section 5.6.

5.5.2 EVALUATION FOR NEW/OCCASIONAL USERS

After acquiring all the user profile and transaction data from the user evaluation of the system, we were able to test the spatial features retrieved for a new/occasional user having no existing user profile stored on the server. To display spatial features most commonly retrieved by like-minded users of the system, the new/occasional users are required to specify the user category that best describes them. The system then examines the spatial features in the transactions table (Table 4.6) corresponding to the selected user category, that contain the highest and second highest frequency of occurrence. It then uses the Apriori algorithm for association rule mining to determine the probability of the target user being interested in every other spatial feature, provided that they were highly interested in the two spatial features with the highest and second highest frequency of occurrence established previously. All spatial features resulting from the association rules with high relevance are then displayed as a personalised map sent to the target user.

Based on the user transaction data acquired from the system evaluation, the following spatial features, shown in Table 5.6, were retrieved for each user category selected. The minimum support value to find the association rules was adjusted to 0.5, whereas the minimum confidence value was set at 0.6. Using these minimum threshold values provides satisfactory relationships between spatial features. If the support value is set too low, a few meaningless association rules are generated in the process, and when the support value is set too high, a lot of meaningful association rules are omitted. The user's transaction data can be found in Appendix A of this thesis.

Table 5.6 - Spatial Features Retrieved for Occasional Users

Features	User Category		
	Tourist	**Surveyor**	**Local Resident**
Normal Roads / Streets	✓ (Second Highest Occurrence)	✓ (Highest Occurrence)	✓ (Highest Occurrence)
Main Roads	✓ (Highest Occurrence)	•	✓ (Second Highest Occurrence)
State Highways	•	✓ (Second Highest Occurrence)	•
Motorways	•	•	✓
Petrol Stations	•	•	✓
24/7 Petrol Stations	•	•	✓
Post Shops	✓	•	✓
Police Stations	•	•	•
Shopping Malls	✓	•	•
Ambulances	•	•	•
Airports	•	•	•

The spatial features retrieved for the three different user categories seem to be quite reasonable and conceivable. The user category ‘Tourist’ has the spatial features normal roads, main roads, shopping malls, and post shops presented, which is likely to be needed by tourists most often rather than the other available spatial features. However, with the incorporation of more spatial features into the system, we can provide a much better and detailed perception of the spatial feature preferences of the various user categories.

5.6 SYSTEM PERFORMANCE SURVEY RESULTS

The results of the system performance survey presented to the users at the end of the system evaluation enabled us to gain an insight into what people generally thought about our system and its functionality. Eleven users were selected to perform the evaluation test of the system. After completion they were each given the short system performance survey rating the performance of various aspects of the system on a scale from 1 to 7, where 1 was 'Extremely Poor' performance, and 7 was 'Excellent' performance. Users were requested to rate the system on the following aspects:

1) The functionality of particular system features
2) The easy usability of the system, i.e., how easy it was to use the features of the GUI, and the overall layout of the GUI
3) The time taken for the system to carry out the user map requests, and the legibility and attractiveness of the displayed map and its features on the mobile device screen
4) The ability of the system to retrieve the spatial features of most relevance to the user, after a few sessions with the GUI

The results obtained from this system performance survey were very positive and have been summarised in Table 5.7 below. The numbers in each cell represents the number of users who rated a particular aspect of the system (specified to the extreme left at the beginning of the row) as the rating in the respective column heading.

Table 5.7 - Summary of System Performance Survey Results

Question 1: Functionality of System Features

	Extremely Poor	Poor	Fair	Average	Good	Very Good	Excellent	N/A
Panning map						4	7	
Zoom in/out					1	3	7	
Routing					2	5	4	
Enable/disable spatial features						4	7	

Question 2: Easy Usability of the System

	Extremely Poor	Poor	Fair	Average	Good	Very Good	Excellent	N/A
Easy Usability						5	6	

Question 3: Time Factor and Map Legibility

	Extremely Poor	Poor	Fair	Average	Good	Very Good	Excellent	N/A
Time factor					3	6	2	
Map legibility					1	7	3	

Question 4: Retrieval of Spatial Features based on User's Past System Usage

	Extremely Poor	Poor	Fair	Average	Good	Very Good	Excellent	N/A
Spatial feature retrieval						4	7	

As seen from the above survey results, all users stated that the spatial features presented to them after a few sessions of testing of the system were extremely relevant to their spatial feature interests, and that the system was accurately able to infer their current spatial feature preferences and requirements, based on their current and past transactions. As depicted in Table 5.7, out of the eleven users selected, seven users rated the feature as ‘Excellent’ and four users rated it as ‘Very Good’. Regarding the easy usability of the system, 55% of the users rated it at ‘Excellent’, whereas the rest 45% rated it at ‘Very Good’.

Two other important features assessed through this system performance survey were the time factor to perform user requests and the legibility of the maps displayed on mobile devices. For the time factor, 18% of users rated it as ‘Excellent’, 55% users rated it as ‘Very Good’, and the rest 27% of users rated it as ‘Good’. Users were also mostly pleased with the legibility of the maps displayed on mobile devices; 27% users rating it at ‘Excellent’, 64% rated it at ‘Very Good’, and 9% users rating it as ‘Good’.

Overall, the system evaluation has revealed some very positive results in all aspects of the system assessed, and the users that carried out the evaluation were very satisfied with the system and its functionalities.

5.7 System Interoperability with Other Mobile GIS

One of the main features of our system is its interoperability with other mobile GIS. Due to our system using open and standard system development tools and standard network communication protocols, it enables the possibility of being able to easily adapt to other mobile GIS that follow similar standards.

The developed GUI has been programmed in Java; hence it offers us platform independence [39], with the ability to migrate our application to practically any platform where a Java Virtual Machine (JVM) exists. According to [15], by the end of the decade, the number of Java-enabled devices in the world is expected to reach over 3.5 billion, and the number of Java developers to reach beyond 10 million. That number is constantly increasing, as having Java becomes more of a necessity than a choice in today’s environment where it is needed for an extensive number of websites and applications. Therefore, our system can be accessed by any laptop or desktop PC, as well as a vast majority of Java-enabled mobile devices available today.

Using standard network communication protocols such as WAP and HTTP enables easy interoperability with other mobile GIS that use similar network communication protocols. Another mobile GIS’s database server can easily be accessed, and in turn our system data can be utilised by those mobile GIS without the need for excessive modifications or alterations.

5.8 CONCLUSION

On a concluding note, the evaluation of our system was successfully carried out with a group of eleven selected users, and has revealed remarkable results in terms of data reduction, as well as retrieval of relevant spatial features based on user preferences. On average, the amount of data reduction achieved in a typical personalised map, based on a sample of personalised maps collected from the system evaluation, is about 28.8%. This reduction in geo-spatial data would in turn affect the time taken, as well as the cost to transfer the data across wireless mobile networks and the Internet. This data reduction proves to be quite crucial for operating the system in an efficient manner, especially when the GUI is being accessed by mobile devices with low data transfer speeds. Another advantage resulting from the reduction in geo-spatial data sent to the user is the increase in map legibility by reducing the overload of potentially unwanted information on mobile devices. This is very vital as everyday mobile devices have quite small display screen sizes, and a map displaying too many spatial features can clutter up the screen, reducing the visibility of the desired spatial features.

When evaluating the system in terms of speed and time to process requests, the average total time taken to process a typical user request was calculated to be at 4.15 seconds. This was broken into three areas, which were the time taken to transfer a user's map request to the server, the time for the server to process the map request and render a map, and the time taken to transfer the map as a response back to the client. The time calculated was based on the time taken to transfer an average map of size 17.8KB to the GUI, and some processing times data obtained from the application server after a certain system evaluation. This value is reasonable for a mobile GIS based on client-server system architecture, as the users conducting the system evaluation had no complaints and seemed satisfied with the easy usability and time based performance of the system.

The evaluation of the combined map personalisation approach also reflected satisfactory results, for both regular users as well as occasional users. All users claimed they were satisfied with the spatial features presented to them after using the system a few times, and that the spatial features reflected their present interests or preferences. Most users rated the easy usability of the system, time to transfer the personalised maps, and the legibility of maps on mobile devices, quite favourably as well. On the other hand, for the occasional users of the system, results obtained regarding the spatial feature personalisation for the three user categories seemed to be relevant for the particular user category. Spatial features retrieved for tourists were normal roads, main roads, post shops and shopping malls; for surveyors they were only normal roads and state highways, whereas for local residents, the spatial features retrieved were normal roads, main roads, motorways, petrol stations, 24/7 petrol stations, and post shops.

CHAPTER - 6

CONCLUSIONS AND FURTHER WORK

6.1 SUMMARY OF WORK COMPLETED

In this thesis, an effective and efficient framework for the personalisation of maps based on the spatial feature preferences of users has been implemented into a working system, using open and standard system development tools. Some of the major accomplishments of this thesis are outlined below.

6.1.1 MAP PERSONALISATION

To personalise maps based on the spatial feature preferences of users, a combination of two information filtering techniques, known as content-based and collaborative filtering, have been researched and implemented into our system. Content-based filtering techniques have been used for regular users of the system who have their user profile stored on the server, whereas collaborative filtering techniques have been used for new/occasional users having no user profile present on the server. The spatial feature preferences of regular users of the system can be collected by implicitly monitoring the spatial features retrieved each session. This data can then be analysed and a user's current spatial feature interests or preferences can be determined, so that spatial features of probable interest to the user can be retrieved the next time they use the system. Collaborative filtering techniques, on the other hand, provide the means of personalising the spatial feature content in maps for a target user having no user profile present, based on the spatial feature preferences of existing like-minded users of the system. Proposing a combined approach to map personalisation, implementing content-based and collaborative filtering techniques, eliminates the disadvantages arising from the independent implementation of either technique. Content-based filtering cannot be feasibly used to retrieve spatial feature content for users having no user profile record stored, and collaborative filtering cannot provide detailed information filtering through contents without having any existing user profile data to examine.

In our system we implement association rule mining to provide collaborative filtering to new/occasional users. Association rule mining is used to discover interesting associations and relations between various spatial features commonly retrieved together by users belonging to the same user category.

Association rule mining is a well known and researched large data mining technique and has been used quite extensively in commercial collaborative recommendation systems. There currently exist various algorithms to mine association rules of interest, such as Apriori, Eclat, and FP-Growth. The best known and currently the most widely used algorithm to mine association rules, the Apriori algorithm, has been proposed to determine the probability of various spatial features being commonly retrieved together by a particular category of users. The reasons for choosing the Apriori algorithm are, i) the ease of implementation compared to other data mining algorithms, ii) its speed of mining association rules is very fast due to the Apriori property, and in most cases finds association rules in linear time, and iii) it can be used to mine association rules in relatively large databases. Adding the implementation of a proven algorithmic solution, such as the Apriori algorithm for association rule mining, to collaborative filtering techniques, helps provide relevant spatial features to new/occasional users of the system, based on the preferences of other existing users.

This combined approach to the personalisation of maps based on spatial feature preferences has been further incorporated into a working system developed using open and standard system development tools. The architecture of this system has been structured based on a four-tier, distributed client-server architecture.

6.1.2 GRAPHICAL USER INTERFACES

To provide an interface for accessing the system, GUIs have been developed for mobile device users as well as desktop PC users. These GUIs have been programmed in Java and HTML to offer platform independency, and utilise standard communication protocols, such as WAP and HTTP, to provide a means of communication between the application and map database servers, which in turn, makes the system interoperable with other mobile GIS. Programming the GUIs in Java and HTML, and using standard communication protocols, also make it possible for average everyday low-cost mobile devices to access the system, anywhere and anytime.

The GUI for desktop PCs was initially developed as a means of testing the system and providing administrative access to its various features. The GUI for mobile devices has been developed using JSP technology, which is essentially an HTML file that allows Java code and certain pre-defined actions to be embedded into static HTML content. The GUI developed for mobile device users has been tailored to meet the requirements of mobile devices' small screens, and due to the varied memory restrictions on mobile devices, the processing of map requests are performed on the server. Some of the main features of the GUI developed for mobile devices include:

- Standard map navigation features, plotting routes, and getting a user's current location through GPS
- Toggling on/off particular spatial features of interest
- The creation of additional interfaces to support the personalisation of maps. This involved allowing regular users to sign in to the system, new users to create and store their user profile on the server, or occasional users to simply select their user category and access the system

6.1.3 SYSTEM EVALUATION RESULTS

A system evaluation was carried out, towards the end of the development stage, by a group of eleven selected users with varied skills and age. The results of the evaluation are very satisfying and comply with the expectations prior to the evaluation. In terms of data reduction, a set of five personalised maps were randomly chosen from the system evaluation sessions, which when compared to the same fully detailed map, resulted in an average data size reduction of about 28.8%, and a 60% average reduction in the number of spatial features retrieved. The reduction in the data size can decrease the time taken as well as the cost to transfer the geo-spatial data across costly wireless mobile networks and the Internet. This proves to be quite crucial when accessing the system through mobile devices with slow data transfer speeds. The reduction in the number of features retrieved dramatically improves the legibility of maps, by reducing the overload of potentially unwanted information. For the small screens of mobile devices, this proves to be very vital and useful, as a map displaying too many spatial features clutters up the screen and may decrease the visibility of certain desired spatial features.

In terms of the performance of the GUI based on time, the average total time to process a user's typical mapping request was found to be 4.15 seconds, as detailed in Section 5.4. This total time is made up of, i) the time taken to send the user's map request to the server, ii) the time taken to process the map request and render the map on the server, and iii) the time taken to send the map response from the server to the client GUI. This time value of 4.15 seconds seems to be quite reasonable for a mobile GIS based on a distributed, client-server architecture, as the user's conducting the system evaluation were very pleased with the easy usability and performance of the system.

Evaluating the combined map personalisation approach also yielded satisfactory and likely results for users creating user profiles and using the system, as well as for occasional users with no desire to store their user profile. After performing the system evaluation, each user was requested to fill out a system performance survey. All users claimed they were very satisfied with the spatial features offered to them after a few sessions of using the system, and that the retrieved spatial features were representative of their present interests or preferences. Users were also asked to rate the easy usability of the system, time to transfer the personalised maps, and the legibility of maps on mobile devices; to which most users indicated they were pleased with the performance of those features of the system. Obtaining user profiles and user transaction data for the different user categories made it possible to infer spatial feature preferences for a target user of the same user category. The results obtained regarding the spatial features retrieved for the three user categories, if no user profile was stored, seemed to be relevant for the respective user category. Spatial features retrieved for tourists were normal roads, main roads, shopping malls and post shops; for surveyors, they were only normal roads and state highways; whereas for local residents, the spatial features retrieved were normal roads, main roads, motorways, petrol stations, 24/7 petrol stations, and post shops.

6.2 FURTHER WORK

Below are a few possible suggestions for further work that could branch out from this research topic.

Personalisation Improvements

Map personalisation currently offered in our system could be improved by allowing user interaction with specific spatial objects on the map, such as a particular shopping mall, or petrol station. This would be useful to learn the user’s social interests, hence being able to recommend similar locations within the vicinity of the user’s current location. This would, in turn, make it possible to recommend particular locations or landmarks to a new or occasional user having similar interests or preferences compared to regular users of the system.

Client J2ME Application

The architecture of the implemented system designed consists of a database server, application server, and client mobile devices. The database server consists of the spatial data for the maps and the application server deploys the GUI accessed by mobile devices through their web browsers.

Instead of using mobile devices' web browsers to access the GUI developed, it would be better to develop an application in the J2ME environment so that the application could be installed onto the mobile device. This would make it easier for the application to communicate to the phone's hardware such as the GPS receiver, and also is more practical for the user to open up an application installed on their mobile device, rather than accessing it through their web browser. The user profiles used for map personalisation could consequently be stored on the mobile device itself, rather than administering them on the server.

Voice Instructions Recital and Recognition

The current GUI for mobile devices could be further developed to provide voice instructions recital and recognition to users through their mobile device's microphone and speaker. Voice instruction recital would be useful for personal navigation, as the user does not have to be constantly looking at the mobile device, and can just follow instructions recited to drive or walk to the selected destination. Whereas, voice instruction recognition can offer further convenience to the user by simply reciting some of the frequently used functions or features available in the GUI, such as plotting a route from location to destination, enabling/disabling spatial features, zooming in/out of the map, etc.

REFERENCES

[1] A. Bookwala, H. Zhou, A. M. Al-Jumaily and J. Collins. "Mobile Personalised Geo-Spatial Applications," *In the proceedings of the Electronics New Zealand Conference 2008*, Auckland, New Zealand, pp. 68-73, November 24-25, 2008.

[2] A. Breuer, J. Hursey, T. Stroman and A. Verma. (2008). "Visualization of Criminal Activity in an Urban Population," *Artificial Crime Analysis Systems: Using Computer Simulations and Geographic Information Systems*, pp. 35-49.

[3] A. Pietracaprina. (2006, June 15). *Introduction to Data Mining*. Unpublished lecture notes, University of Padova, Padua, Italy.

[4] A. R. Walker, B. Pham and M. Moody. "Spatial Bayesian Learning Algorithms for Geographic Information Retrieval," *In the proceedings of the 13th Annual ACM International Workshop on Geographic Information Systems*, Bremen, Germany, pp. 105-114, November 4-5, 2005.

[5] B. De Carolis, G. Cozzolongo and S. Pizzutilo. "Generating Personalized Tourist Map Descriptions," *In the 18th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, Bari, Italy, pp. 259-268, June 22-25, 2005.

[6] B. Kavsek and N. Lavrac. (2006). "Apriori-SD: Adapting Association Rule Learning to Subgroup Discovery," *Applied Artificial Intelligence: An International Journal*, Vol. 20, No. 7, pp. 543-583.

[7] B.-M. Kim, Q. Li, C.-S. Park, S.-G. Kim and J.-Y. Kim. (2006). "A New Approach for Combining Content-Based and Collaborative Filters," *Journal of Intelligent Information Systems*, Vol. 27, No. 1, pp. 79-91.

[8] B. Mobasher, H. Dai, T. Luo and M. Nakagawa. "Effective Personalization Based on Association Rule Discovery from Web Usage Data," *In the proceedings of the 3rd International Workshop on Web Information and Data Management*, Atlanta, USA, pp. 9-15, November 9, 2001.

[9] B. Mobasher, H. Dai, T. Luo and M. Nakagawa. (2002). "Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization," *Data Mining and Knowledge Discovery*, Vol. 6, No. 1, pp. 61-82.

[10] B. Schmidt-Belz, A. Nick, S. Poslad and A. Zipf. "Personalized and Location-Based Mobile Tourism Services," *In the proceeding of the 4th International Symposium on Mobile Human-Computer Interaction*, Pisa, Italy, September 18-20, 2002.

[11] B. Sarwar, G. Karypis, J. Konstan and J. Riedl. "Item-Based Collaborative Filtering Recommendation Algorithms," *In the proceedings of the 10th International Conference on World Wide Web*, Hong Kong, pp. 285-295, May 1-5, 2001.

[12] B. Temesgen, M. U. Mohammed and T. Korme. (2001). "Natural Hazard Assessment Using GIS and Remote Sensing Methods, with particular reference to the Landslides in the Wondogenet Area, Ethiopia," *Physics and Chemistry of the Earth, Part C: Solar, Terrestrial & Planetary Science*, Vol. 26, No. 9, pp. 665-675.

[13] C.-H. Lee, C.-R. Lin and M.-S. Chen. "On Mining General Temporal Association Rules in a Publication Database," *In the proceedings of the 2001 IEEE International Conference on Data Mining*, San Jose, USA, pp. 337-344, November 29 – December 2, 2001.

[14] C. Kyprianidou. (2003). *Analysing Basic Genetics Using Bayesian Networks and the Impact of Genetic Testing on the Insurance Industry*. Unpublished master's thesis, City University, London, United Kingdom.

[15] Conversay: Speech Technology Solutions. (2008). *Background and Motivation: JSAPI2*. Retrieved September 2, 2009, from http://www.conversations.com/JSAPI2JSR113/BackgroundMotivation/tabid/211/Default.aspx

[16] C. Rinner and M. Raubal. (2004). "Personalized Multi-Criteria Decision Strategies in Location-Based Decision Support," *Journal of Geographic Information Sciences*, Vol. 10, No. 2, pp. 149-156.

[17] C. Winkelholz and C. Schlick. "Statistical variable length Markov chains for the parameterization of stochastic user models from sparse data," *In the 2004 IEEE International Conference on Systems, Man and Cybernetics*, Vol. 2, The Hague, Netherlands, pp. 1770-1776, October 10-13, 2004.

[18] D. Gamberger, N. Lavrac and V. Jovanoski. "High Confidence Association Rules for Medical Diagnosis," *In the proceedings of the 4th Workshop on Intelligent Data Analysis in Medicine and Pharmacology*, Washington D.C, USA, pp. 42-51, November 6, 1999.

[19] D. Kelly and N. J. Belkin. "Reading Time, Scrolling and Interaction: Exploring Implicit Sources of User Preferences for Relevance Feedback," *In the proceedings of the 24th Annual International Conference on Research and Development in Information Retrieval (SIGIR '01)*, New Orleans, USA, pp. 408-409, September 9-12, 2001.

[20] D. M. Kristol. (2001). "HTTP Cookies: Standards, Privacy, and Politics," *ACM Transactions on Internet Techology (TOIT)*, Vol. 1, No. 2, pp. 151-198.

[21] E. Mac Aoidh, A. Koinis and M. Bertolotto. "Improving Archaeological Heritage Information Access Through a Personalised GIS Interface," *In the 6th International Symposium on Web and Wireless Geographical Information Systems*, Hong Kong, pp. 135-145, December 4-5, 2006.

[22] E. Mac Aoidh, M. Bertolotto and D. Wilson. "Implicit Profiling for Contextual Reasoning About Users' Spatial Preferences," *In the proceedings of the 2nd International Workshop on Case Based Reasoning and Context-Awareness (held in conjunction with the 7th International Conference on Case Based Reasoning)*, Belfast, Northern Ireland, August 15, 2007.

[23] E. Mac Aoidh, M. Bertolotto and D. C. Wilson. "Understanding Geospatial Interests by Visualising Map Interaction Behaviour," *In the Workshop at the AGILE 2008 Conference*, Girona, Spain, May 5, 2008.

[24] F. M. Facca and P. L. Lanzi. (2005). "Mining Interesting Knowledge from Weblogs: a Survey," *Data and Knowledge Engineering*, Vol. 53, No. 3, pp. 225-241.

[25] F. Mueller and A. Lockerd. "Cheese: Tracking Mouse Movement Activity on Websites, a Tool for User Modeling," *In the proceedings of the*

Conference on Human Factors in Computing Systems (CHI '01), Seattle, USA, pp. 279-280, March 31 – April 5, 2001.

[26] G. Linden, B. Smith and J. York. (2003). "Amazon.com Recommendations: Item-to-Item Collaborative Filtering," *IEEE Internet Computing*, Vol. 7, No. 1, pp. 76-80.

[27] Goldfinger HighTech Ltd. (n.d.). *Goldfinger HighTech Ltd – About Us*. Retrieved May 18, 2009, from http://www.goldfinger.co.nz/profile.htm

[28] Google Mobile. (2010). *Google Maps for Mobile*. Retrieved May 16, 2010, from http://www.google.com/mobile/maps/

[29] iMapia. (2007). *Driving Directions and Maps on your cell phone*. Retrieved May 16, 2010, from http://www.imapia.com/

[30] J. B. Hurwitz. "Empirical Evaluation of Content-Based Filtering for Personalization," *In the proceedings of the 20th International Symposium on Human Factors in Telecommunication*, Sophia-Antipolis, France, March 20-23, 2006.

[31] J. B. Schafer. (2005). The Application of Data-Mining to Recommender Systems. In J. Wang (Ed.), *Encyclopedia of Data Warehousing and Mining* (pp. 44-48). Hershey, Pennsylvania: Idea Group Reference.

[32] J. Barnat, L. Brim and J. Chaloupka. "Parallel Breadth-First Search LTL Model-Checking," *In the proceedings of the 18th IEEE International Conference on Automated Software Engineering (ASE'03)*, Montreal, Canada, pp. 106-115, October 6-10, 2003.

[33] J. Chen, J. Jiang and A. G.-O. Yeh. (2004). "Designing a GIS-Based CSCW System for Development Control with an Event-Driven Approach," *Photogrammetric Engineering and Remote Sensing*, Vol. 70, No. 2, pp. 225-233.

[34] J. Delaney and K. Van Niel. (2007). *Geographical Information Systems: An Introduction, Second Edition*. South Melbourne, Victoria, Australia: Oxford University Press.

[35] J. Doyle, J. Weakliam, M. Bertolotto, D. Lynch and D. Wilson. (2007). "Personalised maps in multimodal mobile GIS," *International Journal of Web Engineering and Technology*, Vol. 3, No. 2, pp. 196-216.

[36] J. Doyle, Q. Han, J. Weakliam, M. Bertolotto and D. Wilson. "Developing Non-proprietary Personalized Maps for Web and Mobile Environments," *In*

the 4th International Workshop on Web and Wireless Geographical Information Systems, Goyang, South Korea, pp. 181-194, November 26-27, 2004.

[37] J. Goecks and J. Shavlik. "Learning Users' Interests by Unobtrusively Observing Their Normal Behavior," *In the proceedings of the 5th International Conference on Intelligent User Interfaces*, New Orleans, USA, pp. 129-132, January 9-12, 2000.

[38] J. Han, J. Pei, Y. Yin and R. Mao. (2004). "Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach," *Data Mining and Knowledge Discovery*, Vol. 8, No. 1, pp. 53-87.

[39] J. Horgan, J. Power and J. Waldron. "Platform Independent Dynamic Java Virtual Machine Analysis: the Java Grande Forum Benchmark Suite," *In the proceedings of the 2001 Joint ACM-ISCOPE Conference on Java Grande*, Palo Alto, USA, pp. 106-115, June 2-4, 2001.

[40] J. J. Sandvig, B. Mobasher and R. Burke. "Robustness of Collaborative Recommendation Based On Association Rule Mining," *In the proceedings of the 2007 ACM Conference on Recommender Systems*, Minneapolis, USA, pp. 105-112, October 19-20, 2007.

[41] J. L. Herlocker, J. A. Konstan and J. Riedl. "Explaining Collaborative Filtering Recommendations," *In the proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work*, Philadelphia, USA, pp. 241-250, December 2-6, 2000.

[42] J. Trajkova and S. Gauch. "Improving Ontology-Based User Profiles," *In the proceedings of RIAO 2004: Coupling Approaches, Coupling Media and Coupling Languages for Information Retrieval*, Avignon, France, pp. 380-389, April 26-28, 2004.

[43] J. Weakliam, D. Lynch, J. Doyle, H. M. Zhou, E. Mac Aoidh, M. Bertolotto and D. Wilson. "Managing Spatial Knowledge for Mobile Personalized Applications," *In the 9th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, Melbourne, Australia, pp. 329-335, September 14-16, 2005.

[44] J. Weakliam, D. Lynch, J. Doyle, M. Bertolotto and D. Wilson. "Delivering Personalized Context-Aware Spatial Information to Mobile Devices," *In the*

5th International Workshop on Web and Wireless Geographical Information Systems, Lausanne, Switzerland, pp. 194-205, December 15-16, 2005.

[45] J. Weakliam, M. Bertolotto and D. Wilson. "Implicit Interaction Profiling for Recommending Spatial Content," *In the proceedings of the 13th Annual ACM International Workshop on Geographic Information Systems*, Bremen, Germany, pp. 285-294, November 4-5, 2005.

[46] K. Mehlhorn and U. Meyer. "External-Memory Breadth-First Search with Sublinear I/O," *In the proceedings of the 10th Annual European Symposium on Algorithms*, Rome, Italy, pp. 723-735, September 17-21, 2002.

[47] L. M. Quiroga and J. Mostafa. "Empirical Evaluation of Explicit versus Implicit Acquisition of User Profiles in Information Filtering Systems," *In the proceedings of the 4th ACM Conference on Digital Libraries*, Berkeley, USA, pp. 238-239, August 11-14, 1999.

[48] T. Landspurg. (n.d.). *J2ME Map*. Retrieved May 16, 2010, from http://j2memap.landspurg.net/

[49] M. Deshpande and G. Karypis. (2004). "Selective Markov models for predicting Web page accesses," *ACM Transactions on Internet Technology (TOIT)*, Vol. 4, No. 2, pp. 163-184.

[50] M. Delgado, M. J. Martin-Bautista, D. Sanchez and M. A. Vila. "Mining Text Data: Special Features and Patterns," *In the proceedings of the ESF Exploratory Workshop on Pattern Detection and Discovery*, London, United Kingdom, pp. 175-186, September 16-19, 2002.

[51] M. Eirinaki. (2005). *Web Mining: A Roadmap*. Technical Report, DB-NET.

[52] M.-H. Tsou. (2004). "Integrating Web-Based GIS and Image Processing Tools for Environmental Monitoring and Natural Resource Management," *Journal of Geographical Systems*, Vol. 6, No. 2, pp. 155-174.

[53] M. J. Zaki, S. Parthasarathy, M. Ogihara and W. Li. "New Algorithms for Fast Discovery of Association Rules," *In the proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining*, Newport Beach, USA, pp. 283-286, August 14-17, 1997.

[54] M. J. Zaki. (2001). "SPADE: An Efficient Algorithm for Mining Frequent Sequences," *Machine Learning Journal*, Vol. 42, Nos. 1-2, pp. 31-60.

[55] M. P. Woodward and R. J. Wood. "Using Bayesian Inference to Learn High-Level Tasks from a Human Teacher," *In the proceedings of the International*

Conference on Artificial Intelligence and Pattern Recognition, Orlando, USA, pp. 138-145, July 13-16, 2009.

[56] M. Raubal and C. Rinner. "Multi-Criteria Decision Analysis for Location Based Services," *In the 12th International Conference on Geoinformatics*, Gävle, Sweden, pp. 47-53, June 7-9, 2004.

[57] M. Worboys and M. Duckham. (2004). *GIS: A Computing Perspective, Second Edition*. Boca Raton, Florida, USA: CRC Press.

[58] N. Jabeur and B. Moulin. "A Multiagent-Based Approach for Progressive Web Map Generation," *In the 2005 OTM Workshop: On the Move to Meaningful Internet Systems*, Agia Napa, Cyprus, pp. 99-108, October 31 – November 4, 2005.

[59] Oracle Technology Network. (2009, July). *Oracle Fusion Middleware MapViewer*. Retrieved May 15, 2009, from http://www.oracle.com/technology/products/mapviewer/index.html

[60] Oracle Technology Network. (2009, August). *Oracle JDeveloper – Official Home Page*. Retrieved May 15, 2009, from http://www.oracle.com/technology/products/jdev/index.html

[61] P. E. Hart, N. J. Nilsson and B. Raphael. (1968). "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," *IEEE Transactions on Systems, Science and Cybernetics*, Vol. 4, No. 2, pp. 100-107.

[62] P. Melville, R. J. Mooney and R. Nagarajan. "Content-Boosted Collaborative Filtering for Improved Recommendations," *In the proceedings of the 18th National Conference on Artificial Intelligence*, Edmonton, Canada, pp. 187-192, July 28 – August 1, 2002.

[63] P.-N. Tan, M. Steinbach and V. Kumar. (2005). Association Analysis: Basic Concepts and Algorithms. In *Introduction to Data Mining* (pp. 327-414). Addison Wesley, US Edition.

[64] P. Pecka. "Software library for the Markov analysis of complex queuing models," *In the 2nd EuroNGI Workshop on New Trends in Modelling, Quantitative Methods and Measurements*, Aveiro, Portugal, November 24-25, 2005.

[65] Q. N. Nguyen, F. Ricci and D. Cavada. "Critique-based Recommendations for Mobile Users: GUI Design and Evaluation," *In the proceedings of the 3rd International Workshop on HCI in Mobile Guides, in conjunction with the*

6th International Conference on Mobile Human-Computer Interaction, Glasgow, Scotland, September 13, 2004.

[66] R. Agrawal and R. Srikant. "Fast Algorithms for Mining Generalized Association Rules," *In the proceedings of the 20th International Conference on Very Large Databases (VLDB'94)*, Santiago, Chile, pp. 487-499, September 12-15, 1994.

[67] R. Agrawal and R. Srikant. "Mining Generalized Association Rules," *In the proceedings of the 21st International Conference on Very Large Databases*, Zürich, Switzerland, pp. 407-419, September 11-15, 1995.

[68] R. Agrawal, T. Imielinski and A. N. Swami. "Mining Association Rules between Sets of Items in Large Databases", *In the proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, Washington, D. C., pp. 207-216, May 26-28, 1993.

[69] R. Burke. (2000). "Knowledge-Based Recommender Systems," In A. Kent (ed.), *Encyclopedia of Library and Information Systems*, Vol. 69, Supplement 32. New York: Marcel Dekker.

[70] R. Castelo, A. Feelders and A. Siebes. "Mambo: Discovering Association Rules Based on Conditional Independencies," *In the proceedings of the 4th International Symposium on Advances in Intelligent Data Analysis*, Lisbon, Portugal, pp. 289-298, September 13-15, 2001.

[71] R. J. Mooney and L. Roy. "Content-Based Book Recommending Using Learning for Text Categorization," *In the proceedings of the 5th ACM Conference on Digital Libraries*, San Antonio, USA, pp. 195-204, June 2-7, 2000.

[72] R. R. Howell. (2008). "On Asymptotic Notation with Multiple Variables," *Technical Report 2007-4.*

[73] S. Brin, R. Motwani and C. Silverstein. "Beyond Market Baskets: Generalizing Association Rules to Correlations," *In the proceedings of the ACM SIGMOD International Conference on Management of Data*, Tucson, USA, pp. 265-276, May 13-15, 1997.

[74] Safe Software: The Spatial ETL Experts. (2009). *The Spatial ETL Experts – Company Overview – Safe Software*. Retrieved May 21, 2009, from http://www.safe.com/aboutus/overview.php

[75] T. Tezuka and K. Tanaka. "Presentation of Dynamic Maps by Estimating User Intentions from Operation History," *In the 13th International Multimedia Modelling Conference on Advances in Multimedia Modeling*, Singapore, pp. 156-165, January 9-12, 2007.

[76] T. Wang and Y. Ren. (2009). "Research on Personalized Recommendation Based on Web Usage Mining Using Collaborative Filtering Technique," *WSEAS Transactions on Information Science and Applications*, Vol. 6, No. 1, pp. 62-72.

[77] TomTom, portable GPS car navigation systems. (2009). *TomTom XL 340•S – All-in-one navigation – TomTom GPS sat nav*. Retrieved May 15, 2010, from http://www.tomtom.com/index.php?Lid=4

[78] UbisWorld.org. (2009, February). *UbisWorld – Research on User Modeling and Ubiquitous Computing*. Retrieved April 24, 2009, from http://www.ubisworld.org/

[79] W. Fangxiong and J. Zhiyong. "Research on a Distributed Architecture of Mobile GIS Based on WAP," *In the proceedings of XXth ISPRS Congress*, Istanbul, Turkey, July 12-23, 2004.

[80] W. Lin, S. Alvarez and C. Ruiz. "Collaborative Recommendation via Adaptive Association Rule Mining," *In the International Workshop on Web Mining for E-Commerce*, Boston, USA, pp. 1-7, August 20, 2000.

[81] X. Niu, R. Ma, T. Ali, A. Srivastava and R. Li. (2004). "On-Site Coastal Decision Making with Wireless Mobile GIS," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 35, No. 2, pp. 1-6.

[82] Y. D. Liang. (2004). *Introduction to Java Programming with JBuilder®, Third Edition*. Upper Saddle River, New Jersey, USA: Pearson Prentice Hall.

[83] Y. Yang and C. Claramunt. "A Flexible Competitive Neural Network for Eliciting User's Preferences in Web Urban Spaces," *In the 11th International Symposium on Spatial Data Handling*, Leicester, United Kingdom, pp. 41-57, August 23-25, 2004.

[84] Y. Yang and C. Claramunt. "A Hybrid Approach for Spatial Web Personalization," *In the 5th International Workshop on Web and Wireless*

Geographical Information Systems, Lausanne, Switzerland, pp. 206-221, December 15-16, 2005.

[85] Y. Zhang, N. Zhang and J. Tang. "A Collaborative Filtering Tag Recommendation System based on Graph," *In the proceedings of ECML PKDD (The European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases) Discovery Challenge 2009*, Bled, Slovenia, pp. 297-305, September 7, 2009.

[86] Yahoo! Mobile. (2010). *Yahoo! Local for mobile*. Retrieved May 16, 2010, from http://mobile.yahoo.com/local

[87] Z. Jrad, M.-A. Aufaure and M. Hadjouni. "A Contextual User Model for Web Personalization," *In the 8th International Conference on Web Information Systems Engineering*, Nancy, France, pp. 350-361, December 3-6, 2007.

[88] Z. Liang and S. Poslad. "A Semantic Enhanced Adaptive Sharable Personalised Spatial Map for Mobile Users," *In the 3rd International Workshop on Semantic Media Adaptation and Personalization*, Prague, Czech Republic, pp. 58-62, December 15-16, 2008.

APPENDIX A – USER TRANSACTION DATA

Transactions for user category **'Surveyor'**:

TRANSID	USERID	USERTYPE	FEATURES
1	3	Surveyor	normal
2	3	Surveyor	normal main state
3	3	Surveyor	normal main state motor
4	3	Surveyor	main state petrol petrol24 post
5	3	Surveyor	normal main state petrol petrol24 airport
6	3	Surveyor	main state shop
7	3	Surveyor	normal main motor shop
8	3	Surveyor	normal main shop
9	3	Surveyor	normal main state motor petrol
1	7	Surveyor	normal state motor
2	7	Surveyor	normal state motor petrol ambulanc
3	7	Surveyor	normal state motor airport
1	11	Surveyor	normal state motor petrol post
2	11	Surveyor	main state petrol post police airport
3	11	Surveyor	normal main state petrol post
4	11	Surveyor	normal main state petrol post
5	11	Surveyor	normal main state petrol post
6	11	Surveyor	normal main state petrol post airport
7	11	Surveyor	normal main state petrol post

Transactions for user category **'Tourist'**:

TRANSID	USERID	USERTYPE	FEATURES
1	1	Tourist	normal main post shop airport
2	1	Tourist	normal main post shop airport
3	1	Tourist	normal main state post shop
4	1	Tourist	normal main shop airport
5	1	Tourist	main state motor post shop airport
6	1	Tourist	normal main post shop
7	1	Tourist	normal main shop
8	1	Tourist	main state motor petrol post
9	1	Tourist	normal main motor shop
10	1	Tourist	normal main state petrol shop
11	1	Tourist	normal main state motor post airport
12	1	Tourist	normal main state shop
13	1	Tourist	normal main motor petrol petrol24 post
1	5	Tourist	main state motor shop ambulanc
2	5	Tourist	normal main state petrol
3	5	Tourist	normal main post airport
4	5	Tourist	normal main state post shop airport
5	5	Tourist	normal main state petrol post
6	5	Tourist	normal main state motor petrol shop ambulanc
7	5	Tourist	normal main post shop airport
8	5	Tourist	main state motor shop airport
9	5	Tourist	normal main state post shop ambulanc
10	5	Tourist	main state motor post
1	8	Tourist	normal main post
2	8	Tourist	normal main motor petrol petrol24 post shop
3	8	Tourist	police shop
4	8	Tourist	main state post shop
5	8	Tourist	main motor petrol post airport
6	8	Tourist	normal state motor post ambulanc airport
7	8	Tourist	main motor petrol petrol24 shop
8	8	Tourist	normal main state motor post shop airport
9	8	Tourist	normal main motor post shop
1	10	Tourist	normal main post
2	10	Tourist	normal main post shop
3	10	Tourist	normal main motor post shop ambulanc
4	10	Tourist	normal main motor petrol post shop
5	10	Tourist	normal main motor post police shop
6	10	Tourist	normal main motor petrol petrol24 post shop
7	10	Tourist	normal main motor post shop

Transactions for user category **'Local Resident'**:

TRANSID	USERID	USERTYPE	FEATURES
1	2	Local Resident	normal main state motor post
2	2	Local Resident	normal main state motor post police
3	2	Local Resident	normal main state post
4	2	Local Resident	normal main state motor post
5	2	Local Resident	normal main state motor post police shop ambulanc
6	2	Local Resident	normal main state post
7	2	Local Resident	normal main state motor post
1	4	Local Resident	normal main post
2	4	Local Resident	normal main petrol petrol24 post
3	4	Local Resident	normal main motor petrol petrol24 post shop
4	4	Local Resident	normal main motor post
1	6	Local Resident	normal main motor petrol petrol24 post
2	6	Local Resident	normal main motor petrol petrol24 post police ambulanc
3	6	Local Resident	normal main petrol petrol24 post
4	6	Local Resident	normal main motor petrol petrol24 post
5	6	Local Resident	normal main motor petrol petrol24 post
6	6	Local Resident	normal main motor petrol petrol24 post airport
7	6	Local Resident	normal main motor petrol petrol24 post police
1	9	Local Resident	normal main motor petrol petrol24 post police airport
2	9	Local Resident	normal main motor petrol petrol24 post police shop airport
3	9	Local Resident	normal main motor petrol petrol24 post police airport
4	9	Local Resident	normal main motor petrol petrol24 post police ambulanc airport
5	9	Local Resident	normal main motor petrol petrol24 post police airport

APPENDIX B – TYPICAL MOBILE MAP DEVICE

A typical currently available proprietary handheld GIS is the TomTom XL 340 [77].

Figure – TomTom XL 340 [77]

The main purpose of this device is to provide in-car navigation for frequent drivers based on pre-installed maps of the user's region. It offers voice guidance directing the driver through a calculated route from a start location to a destination, reducing the risk of the driver being distracted from the road. This route is calculated based on either the shortest distance between two points or least amount of travel time. And with the presence of automatic re-routing, routes are recalculated if the driver goes off the route.

All these features mentioned above exist in most GPS navigators currently available on the market.