WEB GIS APPLICATION IN LOCAL GOVERNMENT: MUNICIPALITY OF GÄVLE CASE STUDY

Alexey V. Tereshenkov

June 2009

Master’s Thesis in Geomatics

MSc Geomatics Programme
Examiner: Anders Östman
Supervisor: S. Anders Brandt
Abstract

This project was aimed at critically analyzing the use of web geographical information system (GIS) applications in local government, and problems associated with the design, development, and implementation of web mapping applications in the case study of the municipality of Gävle, Sweden. The project included a scientific and thorough analysis of options for developing web GIS applications by using ESRI GIS software technologies. During this research project, the ESRI GIS software tools and packages were used to perform a comprehensive system analysis of the proposed web mapping application, to gain a better understanding of the available methods of map document and web GIS application performance optimization, and finally to develop a web mapping application prototype for the web map of the municipality of Gävle. The research results include the guidelines for preparing map documents to be published, optimizing ESRI ArcGIS Server web mapping applications performance, and some possible ways of customizing user tools and a user interface by using the Microsoft Visual Studio framework. This study proves that ESRI web-based GIS products provide viable tools for web mapping application design, development, and customization. The developed ArcGIS Server-based web mapping application can be optionally improved and then be used instead of existing ESRI ArcIMS-based application in the municipality of Gävle.

Keywords: web GIS, ESRI, local government, ArcGIS Server
Preface

The present research project is the result of completing a 1-year Master of Science programme at the University of Gävle, Sweden. This project was carried out within 10 weeks as a Degree project for Master of Science in Geomatics (60 ECTS). Within this project, ESRI GIS software have been used in order to develop a web GIS application at the municipality of Gävle, Sweden. This study has been conducted at the GI-department of the municipality of Gävle as a part of a current workflow in the municipality of Gävle. The results of the performed research offers numerous insights and implications for using GIS to the study of local government workflows, including such topics as web map drawing performance increases, map caching generation and update automation techniques, and web GIS tools development and customization. Thus, this thesis report can be useful to a variety of readers from different disciplines. Specialists interested in municipal GIS design and implementation could find much of significance in this master thesis report. Those readers concerned with the use of web programming and applications of ESRI GIS software in local government might also find much of value in this report. The given report might be useful to academics and nonacademics as well, such as those involved in government GIS workflows and web mapping application development.
Acknowledgements

Many people have contributed either directly or indirectly to this study. This research has been carried out with great support from my supervisor S. Anders Brandt, PhD, who greatly assisted me with refining the structure and content of the report. The municipality of Gävle kindly provided me with the required data and technical guidance. I am also very thankful to the staff of the municipality of Gävle, GI-department; in particular, Fredrik Ekberg and Kaj Waldenby for their technical support and sharing their knowledge during the project work. I am deeply grateful for all the different types of support from the municipality of Gävle GIS Coordinator Annelie Höök and GIS Manager Eddie Larsson.

I am thankful to my colleague Denis Perechnev for his help with programming code in preliminary studies of the research project and his tireless enthusiasm. I would like to thank my good friends Petra Norlund, Tian Jiang, Wigger Tims, and Chris Wilms for their comments on an earlier draft of this manuscript. I am especially grateful to Xintao Liu who made comments on the programming code embedded in the developed web application and provided me with unyielding encouragement. However, all errors, oversights, and shortcomings are entirely my own.
# Table of contents

Abstract .................................................................................................................................................. 2  
Preface .................................................................................................................................................... 3  
Acknowledgements ............................................................................................................................... 4  
List of acronyms ..................................................................................................................................... 6  
1 Introduction .......................................................................................................................................... 7  
   1.1 Background ..................................................................................................................................... 7  
   1.2 Aim and purpose ............................................................................................................................. 7  
   1.3 Delimitations ................................................................................................................................... 8  
2 Use of GIS applications in local government ....................................................................................... 9  
   2.1 Use of GIS in the municipality of Gävle: an overview .................................................................... 10  
   2.2 Types of data and main applications of GIS in the municipality of Gävle .................................... 11  
   2.3 GIS software resources used in the municipality of Gävle .............................................................. 12  
3 Web GIS in local government: literature review ............................................................................... 15  
   3.1 Web GIS applications: concepts and issues .................................................................................... 15  
   3.2 Access to GIS data via the web: concepts ....................................................................................... 17  
   3.3 Efficacy and issues of web GIS applications ................................................................................... 19  
   3.4 Importance and benefits of web GIS applications .......................................................................... 19  
   3.5 Development of web mapping applications: issues and recommendations .................................. 20  
4 Web GIS development in the municipality of Gävle: case study ......................................................... 22  
   4.1 Demand on new web GIS platform in the municipality of Gävle ..................................................... 22  
   4.2 Requirements for the target web mapping application: system analysis ...................................... 23  
   4.3 ArcGIS Server as a platform for local government web GIS applications .................................... 25  
5 Web mapping application development: methodology .................................................................... 27  
   5.1 Map document preparation .............................................................................................................. 28  
   5.2 Publication of the map document as a map service ........................................................................ 29  
   5.3 Caching the map service content .................................................................................................... 29  
   5.4 Creating a web mapping application .............................................................................................. 31  
   5.5 Evaluation of performance of the web mapping application .......................................................... 31  
   5.6 Customization of the web mapping application ............................................................................. 32  
6 Web mapping application development: results .................................................................................. 34  
   6.1 Map document preparation .............................................................................................................. 34  
   6.2 Publication of the map document as a map service ........................................................................ 35  
   6.3 Caching the map service content .................................................................................................... 35  
   6.4 Creating a web mapping application .............................................................................................. 37  
   6.5 Evaluation of performance of the web mapping application .......................................................... 40  
   6.6 Customization of the web mapping application ............................................................................. 40  
7 Discussion ............................................................................................................................................. 42  
8 Conclusion ............................................................................................................................................ 46  
9 Future prospects ..................................................................................................................................... 47  
References ............................................................................................................................................... 49  
Appendix A ................................................................................................................................................ 54  
Appendix B ................................................................................................................................................ 55
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADF</td>
<td>Application Developer Framework</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>ASP</td>
<td>Active Server Pages</td>
</tr>
<tr>
<td>CPU</td>
<td>Central Processor Unit</td>
</tr>
<tr>
<td>DBA</td>
<td>DataBase Administrator</td>
</tr>
<tr>
<td>DBMS</td>
<td>DataBase Management system</td>
</tr>
<tr>
<td>DLL</td>
<td>Dynamic Linked Library</td>
</tr>
<tr>
<td>ESRI</td>
<td>Environmental System Research Institute</td>
</tr>
<tr>
<td>GIS</td>
<td>Geographical Informational System</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>HTML</td>
<td>HyperText Markup Language</td>
</tr>
<tr>
<td>DHTML</td>
<td>Dynamic HyperText Markup Language</td>
</tr>
<tr>
<td>HDD</td>
<td>Hard Disk Drive</td>
</tr>
<tr>
<td>HTTP</td>
<td>HyperText Transfer Protocol</td>
</tr>
<tr>
<td>IIS</td>
<td>Internet Information Services</td>
</tr>
<tr>
<td>ISO</td>
<td>International Standard Organization</td>
</tr>
<tr>
<td>JPEG</td>
<td>Joint Photographic Experts Group</td>
</tr>
<tr>
<td>JSP</td>
<td>Java Server Page</td>
</tr>
<tr>
<td>KML</td>
<td>Keyhole Markup Language</td>
</tr>
<tr>
<td>OGC</td>
<td>Open Geospatial Consortium</td>
</tr>
<tr>
<td>OS</td>
<td>Operating System</td>
</tr>
<tr>
<td>PDF</td>
<td>Portable Document Format</td>
</tr>
<tr>
<td>PNG</td>
<td>Portable Network Graphics</td>
</tr>
<tr>
<td>RAM</td>
<td>Random Access Memory</td>
</tr>
<tr>
<td>RDBMS</td>
<td>Relational DataBase Management System</td>
</tr>
<tr>
<td>ROI</td>
<td>Return On Investment</td>
</tr>
<tr>
<td>RPM</td>
<td>Revolutions per minute</td>
</tr>
<tr>
<td>SLD</td>
<td>Styled Layer Descriptor</td>
</tr>
<tr>
<td>SQL</td>
<td>Structured Query Language</td>
</tr>
<tr>
<td>TCP/IP</td>
<td>Transmission Control Protocol/Internet Protocol</td>
</tr>
<tr>
<td>WCS</td>
<td>Web Coverage Service</td>
</tr>
<tr>
<td>WFS</td>
<td>Web Feature Service</td>
</tr>
<tr>
<td>WMA</td>
<td>Web Mapping Application</td>
</tr>
<tr>
<td>WMS</td>
<td>Web Map Service</td>
</tr>
</tbody>
</table>
1 Introduction

1.1 Background

Many things have changed since the Internet was first implemented. Our information tools and computer systems have changed dramatically. Humans now have access to multiple full-service computer workstations. The Internet has linked a wide variety of human knowledge and provided access to information, which has been accumulating for centuries. The web has penetrated just about every sphere of human interest and using information from the web has become ubiquitous among different categories of users (Battelle, 2005). However, granting information to the public has led to the question of how effectively this information can be served.

The demand for different kinds of knowledge is increasing very fast as well. To date, fast and easy access to various information is assumed to be common practice. A number of applications of geographical information are available nowadays. Geographical information is a type of information about any objects or phenomena, which can be described by a location relative to the Earth surface (New Zealand Geospatial Office, 2009). Huge amount of various geographical data make up the majority of information managed within local governments.

New requirements and possibilities of data access, dissemination, and acquisition stipulated interest in technologies, which are capable of effectively managing geographical information. This has resulted in wide adoption of geographical information systems (GIS) as the de facto standard of geographical data acquisition, storage, and management. Moreover, the presence of a growing number of geodata users – among both citizens and local government – raised a question of the necessity of providing wide public access to geographical data. The broadest access to information is provided by the global computer network – the Internet. It has become a very fundamental part of present information workflows in the Swedish local government as well. In conclusion, the possibility of using the Internet provided the means for the development of efficient, scalable, and useful GIS applications that could satisfy the needs of users of geographical information.

1.2 Aim and purpose

The aim of this research project was to describe the present use of geographical information online in the case study of the municipality of Gävle (Sweden). Thereafter, the study developed a web mapping application prototype with GIS functionality to serve the needs of internal users and city residents.

In the scope of the given research, different technologies used for sharing municipal geoinformation via the Internet have been analyzed; some information on the historical background of web map use in the municipality, lessons, and challenges are briefly reviewed. Some technical issues of geographical data sharing were critically discussed. Disadvantages and benefits of sharing geoinformation via the web by using web GIS applications were briefly covered as well. Finally, a web mapping application was developed and possible ways for improving it are proposed.

This research project was vital because it helped to fill a gap in the formal scholarly literature on applied urban studies and geographical information systems. This study was devoted to the straightforward application of geospatial technologies to the practical use of municipal maps on the web. One motivation for this work was that developing a sound web based GIS application at the local government level could facilitate the access to GIS data by the public and would therefore be beneficial. Another motivation was that a web GIS application implementation would enable a software platform, user-system interaction methods, and GIS application development principles to be tested and compared in the same case study. Based on existing experiences and lessons learned from previous studies, it was possible to seek out one of the most optimal and efficient methods for a specific web mapping application deployment in the municipality of Gävle for the public use.
1.3 Delimitations

Methods from numerous disciplines have been adopted in the research, including, but not limited to, programming, geographical information science, system administration, and web design. Obviously, there is no possible way to include a detailed description of each task that has been performed during the project. Consequently, several topics have been left beyond the scope of the thesis report.

In Chapter 2, a brief definition of GIS has been introduced. A detailed description of GIS components, history of development, and various applications are not included. Local government workflows in the municipality of Gävle are left outside the scope of the research as well. In the literature review chapter (Chapter 3), case studies on implementing web GIS applications in other local governments have not been reviewed. Detailed descriptions of the history of cartographic information publication on the Internet and technical descriptions of the systems are omitted. Detailed description of map document feature attributes are left outside the content of the report.

In Chapter 4, only brief descriptions of the used software products are given. The chosen platform for application development is defined very briefly and other possible platform alternatives are not reviewed. Hardware and network tuning is left outside the scope of this research and only some of the potential paths for the system optimization are given in the future prospects chapter (Chapter 9). Future perspectives on the topic cover mostly technical aspects and do not cover in detail related fiscal or management issues. Citizen and government interaction aspects have not been envisaged in detail. Finally, public participation and e-government concepts are outside the scope of the given thesis work as well since these topics should be learned in more detail from a management perspective that is not possible due to time limit.
2 Use of GIS applications in local government

Geographical information system (GIS) technology is an information technology that utilizes location data to integrate different types of information. It uses a geographical, or geospatial, approach to bring data from numerous sources together and to uncover complex relationships otherwise difficult to understand. GIS affects nearly every aspect of modern local government work and helps provide a foundation for integrating municipal services. Many communities have built their large-scale information system by using this technology. GISs have been recognized as a very important tool for decision support and planning analysis. A number of local government agencies realized the possibility of providing decision-makers with comprehensive and up-to-date information by using GIS (Peng, 2001).

However, the implementation of a municipal GIS for city planning and managing of municipal resources is a very sophisticated as well as a vital issue from both the scientific and production perspective. The process of national data infrastructure development and propagation in many countries around the world can be noticed. Particularly, in Europe spatial data infrastructure (SDI) development has a very important role: the INSPIRE (infrastructure for spatial information in Europe) directive could affect the GIS data management workflows and standards on the city level as well. That implies that providing wide access to GIS data will become even more important in the future and could bring some changes in the local government GIS development (EC-INSPIRE, 2004).

Moreover, as cities grow, the complexity and volume of the spatial databases in municipalities increase. This way, all of these data should be properly collected, used, and maintained with tools of a local government GIS. Moreover, city managers must have access to an appropriate information system that could help them in the decision-making process. Just the same, citizens should have access to map information and geospatial services in order to save their time and facilitate access to geographical data (Huxhold, 1991).

There are a great number of previous studies regarding municipal GIS design and implementation. Works on geoinformation technology for spatial and urban planning are numerous as well and include countless number of manuscripts, textbooks, and scientific articles. This theme is all-important for many municipal agencies from different countries including Sweden. Numerous Swedish research institutes and government sector agencies actively participate in the process of applying GIS methods to the city management. However, there are many versions of the “municipal GIS” definition, its purpose, and structure in the scientific community. These terms can differ in composite elements of the city GIS conceptual model, spatial data models, municipal and legal components hierarchy. Nevertheless, during last decades it has been noticed that using geoinformation technology (i.e., GIS) tools with existing IT systems and enterprise databases can provide a more productive and efficient “geoinformative” environment for city management (Tikunov, 2004).

Basically, using GIS seeks to lead to a simplification of workflows. GIS allows leveraging spatial, or geographical, approach for management of municipal resources. In addition, the ability to register non-spatial, or attributive, information about geographical objects (referred to as features in GIS literature), GIS provides users with the ability to define spatial location geographically, i.e., by using maps instead of text description. GIS is capable of identifying spatial relations between features, comparing their influence on the city development, population growth, criminal activity, and so forth (Tomlinson, 2003). It is important to note that GIS helps integrate data from different city agencies and municipal departments’ information systems in a unified integrated “geomunicipal environment.”

Local governments might employ GIS technologies in many ways. GIS helps city agencies to answer many questions and simplifies the processes of retrieving real estate data, address information, and cartographic views. While dealing with such an amount of data, it is important to develop an adequate, flexible, and simple access to GIS datasets. In almost any case, particularly, while dealing with GIS information in local government, GIS departments’ specialists must provide services to government leaders and other departments’ staff. Furthermore, spatial, or geographical, data often needs to be available to the public audience (Jensen et al., 2005).
As has been acknowledged by Huxhold (1991), municipal structures have a very strong demand for using information systems with rich functionality and advanced capabilities that could support the process of making decisions. However, relatively often GIS and IT managers at the municipal level might be placed in a very limited fiscal environment. Therefore, city GIS specialists would face a peculiar situation: a city could be in need of powerful spatial data management system, but no one is buying data, equipment, and software and paying specialists for their work. In other words, municipalities ask GIS specialists to develop rich-functional information systems, but they are not ready to invest appropriate resources for these projects and are not aware of possible difficulties and obstacles that GIS specialists can be confronted by.

Fortunately, to provide better citizen service and control costs, more Swedish municipalities are starting to realize the need to investment in GIS, because the ability to process, manage, and update large amounts of information efficiently is a worthwhile asset for local governments and their constituencies. That is because GIS tools can fit the enterprise business logic very well and provide good integration of spatial data storage mechanisms with other existing information system interfaces. Whether for community planning, tax assessment, or engineering development, municipalities require fast and easy access to the maps and associated descriptive data in an automated system. Finally, local government needs to be able to retrieve and analyze these datasets to meet increasing service demands (Bertolotto et al., 2001).

However, in this limited resource environment, general methodology principles of citywide geoinformation system design and implementation strategies as a rule are not adequate. Without having appropriate software and hardware support, these systematic project development rules cannot be realized into practice. Therefore, many municipalities are looking for non-traditional geoinformation decisions that could allow satisfying most municipal management needs in an effective and proper way (Campbell & Masser, 1995).

While analyzing literature on the topic of municipal server GIS implementation on the city level, it can be noticed that not so many articles, which cover the given topic, exist. Indeed, methods that could be adopted for implementation of modern city-wide GIS in local government are not covered very well in academia and might be more business-oriented, i.e. oriented to industrial workflows. Nevertheless, research on successful cases of municipal GIS implementation showed that there is no unique and absolutely true path to the successful local government GIS application (Sussman, 1996).

As was defined above, GIS is a very sophisticated type of information system, and GIS methods of data mining and analysis are very complicated from the business process management point of view. In addition, developing a GIS for local government purposes is a complex process, which might require the reconciliation of diverse source materials, conversion of paper maps and other hard copy data, and many others. Acquistion of new data, hardware, and software, and development of user applications are required. In this process, staff of GIS departments could redefine their work order management, redesign workflows, and seek to utilize innovative methods of providing access to GIS data.

The understanding of those issues in local government GIS applications could help in further research upon developing a web GIS application for providing wide access to geographical data. The lessons – of management, fiscal, and spatial data infrastructure aspects – from previous research and business have been shortly reviewed above to elaborate the most important aspects of local government GIS application.

2.1 Use of GIS in the municipality of Gävle: an overview

Municipality of Gävle is a Swedish municipality that is located by the Baltic Sea coast in the east central part of Sweden. The municipal seat is the City of Gävle. Geographically the municipality is situated a little north of the mouth of the river Dalälven; the municipality of Gävle is the southernmost municipality of the land Norrland (Figure 1).
The present municipality was created in 1971; at that time, several rural communes were unified to the one municipality of Gävle. Nowadays, the municipality includes several localities. In 2005, the population of the municipality of Gävle made up around 92,000 people. The highest population in the municipality has such cities as Gävle (around 70,000), Valbo (around 7,000), Forsbacka (around 1,700), Hedesunda and Norsundet (both around 1000) (City of Gävle, 2009).

The department of Geographic Information (GI-department) of the municipality of Gävle provides cartographic support, maintenance, and development of all geographical information for the municipality of Gävle departments within the municipality of Gävle and partly for surrounding municipalities. The GIS staff has been deeply involved in developing both web and stand-alone applications over the last decade. The municipality has a long history of analysis, maintenance, and creation of geographical data within the municipality of Gävle. Its primary function is to provide access to high-quality geographical information services for both internal and external users. The service area is delimited to different base maps creation, custom map products, application development, and consulting other municipalities of the Gävleborg county.

Thus, the GI-department is the body that is responsible for acquisition, storage, and operations with geospatial data for the municipality. To date, the municipality is growing very fast and to facilitate the process of working with geographical datasets both within and beyond the municipality is required. To satisfy these needs, several web GIS applications were developed to assist other departments staff work and address public audience users’ needs. The geographical databases in year 2009 included dozens of gigabytes of information. Huge amount of data are regularly being bought from other government agencies (for instance, the Swedish mapping, cadastral and land registration authority (Lantmäteriet)) as well and appropriate data licensing agreements have been made with GIS industry companies (for example, ESRI Inc. (USA) and Safe Software Inc. (Canada)).

2.2 Types of data and main applications of GIS in the municipality of Gävle

GIS have been used in Gävle since 1988. Main GIS databases include numerous datasets of geographical information, namely:

![Figure 1. Location of the municipality of Gävle on the map of the Gävleborg county and the municipality of Älvkarleby. [Based on cX-Länskarta GIS application; published with permission of the municipality of Gävle]](image)
• Data on companies (registry)
• Real estate data (registry)
• Address data/Buildings data (address, buildings registry)
• Population data (municipality citizens registry)
• Personnel data
• Infrastructure data (communication)
• Schools information (pupils registry, schools)
• Hydrology data (rivers, streams, channels)
• Environment development data
• Detailed city plans data
• City administrative data.

Within the municipal GIS, there are three main types of reference maps (Figure 2):

• Primarkarta (black-and-white detailed municipality-wide map for internal use)
• cX-Karta (country-wide color address map)
• Ortofoto (municipality-wide aerophoto images).

![Figure 2. Primarkarta, cX-Karta, Ortofoto karta (from the left to right)
Based on existing GIS applications; published with permission of the municipality of Gävle.]

The main GIS applications have been designed to satisfy specific needs of internal and external users. Some of them are implemented as desktop GIS; others are developed as web applications:

• cX-Länskarta (base map web mapping application, http://gis.gavle.se/lanskartan)
• FDBcX (web mapping application, real estate database, http://gis.gavle.se/FDBcX)
• BEFO_statistik (web mapping application, country population data http://gis.gavle.se/befostat)
• Planarkivet (web application for viewing applied development plans, http://gis.gavle.se/planer)
• Vattenskydd (application for discovering a position of a water service area - inside/outside http://gis.gavle.se/vattenskydd)
• Skolskjuts (web mapping application for public school districts, http://www.gavle.se/skolskjuts)
• PVP (primary health care agreement web application, http://gis.gavle.se/pvputb)
• STIG (http://gis.gavle.se/stig, web mapping application for report preparation for internal use).

2.3 GIS software resources used in the municipality of Gävle

ESRI GIS software has been in use in the municipality of Gävle since 1988. Over the course of this time, the GIS use has evolved from centralized ArcGIS Desktop software to a decentralized GIS application. As the GI-department grew, a greater number of geodatasets were included in production GIS. As a result, more staff in different municipality departments, including public works, planning, and environment development, began using maps and GIS data in their work as well.

Many projects at the municipality of Gävle typically involve different types of geospatial information – surveying data, remote sensing data, and database management system repositories – that must be collected, analyzed, and visualized. As a rule, such information is stored in many locations: in relational databases, mobile storage devices, maps, etc. To facilitate the management and
access to the information, different software solutions have been built within the municipality and various ESRI GIS products have been used as well. These solutions range from spreadsheet based custom executable applications to huge custom enterprise-level systems.

Over time, the question of whether to transfer most of GIS datasets to a more secured multi-user DBMS-based system has been raised. Consequently, the exploitation of single-use ArcGIS Desktop product has been expanded to the use of ArcSDE – an ESRI GIS product for the storage and manipulation of spatial data in relational database management systems (RDBMS). ArcSDE provides a multi-user data storage framework based on underlying proprietary relational database management system technology (Batcheller et al., 2007). Originally, geodatasets have been stored in shapefiles (ESRI vector data format), personal databases (ESRI database format for raster and vector data storage, which is based on Microsoft Access .mdb format) and file geodatabases (ESRI database format for raster and vector data storage, which is based on Microsoft OS Windows file folder format). However, over the time, all of these datasets have been transferred to ArcSDE geodatabases. This allows efficient storage of spatial datasets, providing the capabilities of multi-user access to data, and many others. As the main DBMS of the municipality of Gävle, Microsoft SQL Server 2005 is managed by ArcSDE. Microsoft SQL Server 2005 provides easy-to-use administration tools, is capable of storing geographical data, and communicates with the ArcSDE platform.

The importance of access to stored geographical information for the public was realized in the municipality of Gävle several years ago. Organization of web access to data within the municipality of Gävle was required due to many reasons. Relatively often web access to GIS information might be required by other agencies or other municipalities, because they do not have software compatible with data formats and standards used in the GI-department at the municipality of Gävle. For instance, several other municipalities in Sweden are users of MapInfo and Autodesk program products, which do not allow them to use ESRI GIS documents easily. For these reasons, publishing Open Geospatial Consortium (OGC) compatible web map services (WMS), web feature services (WFS), web coverage services (WCS) on the Internet has been done within the municipality. In order to provide public access to map data and some information stored in GIS databases to the public over the Internet different software has been used. Originally, ESRI ArcIMS software was used for creating a web mapping applications both for external and internal use (Figure 3). ArcIMS provided a relatively flexible and scalable platform for publication of GIS map documents over the web. Nowadays, in the GI-department, several ArcIMS-based web mapping applications have been implemented.

For customization of ArcIMS-based web GIS applications, a product of ESRI Australia EVie has been used by the GIS staff of the municipality. EVie offers GIS specialists in the GI-department the possibility to implement a customized highly tailored web mapping application that could provide access to GIS data via an relatively easy-to-use web interface. Thus, the use of a web-based GIS gateway in the municipality of Gävle provided ready access to GIS data assuring increased return on investment (ROI) through maximized information utilization, resulting in better informed decision-making, enhanced productivity, and improved citizen service.
Figure 3. cX-Länskartan (base web mapping application for the whole Gävleborg county) [cX-Länskarta web application; available at http://gis.gavle.se/lanskartan].

Nowadays, the department is in the process of migration to the ESRI ArcGIS Server platform for publication of maps and GIS tools over the Internet. ArcGIS Server is another server GIS product from ESRI that provides a platform for connecting GIS users with the geographical information they are in need of by employing web mapping applications and web GIS services. To date, the analysis of ArcGIS Server platform capabilities is being performed by the GI-department. The current matter to the department is to analyze how effectively the ArcGIS Server platform can be used to build fast, high-quality mapping applications, which could substitute existing ArcIMS-based web applications.

In the very moment, an evaluation of the ESRI S-GROUP (Sweden) product GEOSECMA for ArcGIS is being performed. GEOSECMA is a municipal geoinformation system that is based on ArcGIS Server platform. The GI-department expects that the product could help to improve productivity of web GIS, desktop, and mobile workforces. GEOSECMA could help to integrate state-of-the-art GIS analysis and mapping solutions with other municipal GIS components. In addition, GEOSECMA is capable of simplifying the management of geodatabases, services, and mapping applications. Finally, it provides powerful web-platform wizard-based platform for creating GIS-integrated web applications for use in the municipality (ESRI-SGROUP, 2009).
3 Web GIS in local government: literature review

3.1 Web GIS applications: concepts and issues

A discussion of providing access to GIS data is worthwhile due to the wide adoption of the Internet for publishing geodatasets during the last decades. Creating web maps is important and can facilitate access to GIS documents, tools, and data for a wide audience. However, successful web mapping application development in local government requires performing a comprehensive assessment of the needs and requirements of municipalities and citizens.

Within the thesis report, two similar terms have been used – web GIS application and web mapping application. In order to make it clearer to a reader, these terms should be introduced. Web mapping application is the term that was adopted from ESRI ArcGIS Server product terminology and it is a web application that can be accessible via a web browser. It might include a dynamic map, basic map manipulation tools, and simple tools for map data querying – information searching and feature identifying, for instance. Even though web GIS application is similar to web mapping application description, it is rather a broad term. Web GIS application implies presence of GIS tools for analyzing and processing geographical map data. In other words, web mapping application is focused more on viewing map, whereas web GIS application aims to allow users to analyze process map data in order to retrieve new spatial or non-spatial information.

Efficient and sound web mapping solutions should firstly be developed, and then customized and refined through interviews with application users. Based on the user feedback, web mapping developers will be able to assess user functions, GIS objectives, and existing resources, including technical expertise, hardware, software, and data. Based on this information, GIS specialists should evaluate potential web GIS functionality across departments, prioritize needs, and recommend an implementation plan that makes the best use of current assets (Tikunov, 2004).

As noted in the works of researchers (Hansen & Prosperi, 2005; Scharl & Tochtermann, 2007), during GIS application development it is crucial to address such questions as target user skills, application usability, and system performance. All of this information should be documented in a comprehensive report that serves to guide government personnel throughout the development process. However, in the municipality of Gävle, this process has been tested over several years and no staff have high enough skills enough to discover and overcome possible challenges.

As was acknowledged by many specialists (ESRI, 2008) the idea of developing web GIS applications based on a local government GIS can be approved if financial benefits that an agency could meet while integrating GIS in its everyday work could be clearly stated. For instance, transportation companies can discover that by using GIS tools they can deliver more orders for their clients since their staff will be able to analyze the traffic, faster solve routing tasks, and so forth. Hence, company managers will be able to see immediate increases in profits. Thereafter, companies can estimate whether it is beneficial to buy required software, perform training, and pay for technical support. In short, all of these factors, coefficients, and values could be converted to real values and can be interpreted in terms of economic and market indicators.

Web GIS development evaluation, however, is rather a different process. The efficiency of web GIS implementation for local government cannot very often be demonstrated in formalized values. For example, it is very difficult to state that to use customized web mapping application is easier than to use out-of-the-box desktop applications. Just the same, it is almost impossible to register the time savings when city residents use web applications instead of coming to the office. Thus, it is hard to acknowledge the financial benefits of web GIS applications in a short-term perspective (Hansen & Prosperi, 2005).

Nevertheless, during web GIS application development, there are many coefficients and indicators that, actually, could be formally figured in specific values – a decrease of staff hours that are required for task accomplishment, number of involved professionals, and fewer amounts of resources required
for fulfilling a work order. It could take months or even years to register the benefits of using web GIS applications at the enterprise level. Even though proving these points is often very difficult, because not all these benefits could be acknowledged immediately after a web GIS implementation, the efficiency of putting GIS onto the web is widely acknowledged to date and is becoming a de-facto standard for data sharing and map publication (Kraak, 2001).

Numerous local government agencies around the world maintain geographical information on local geographical areas that are required to become available to the public. Previously, to obtain specific information about a geographical object (parcel of land, for example), one had to visit the municipal department office and apply for it on-site. Nowadays, however, thanks to the World Wide Web, this procedure is being changed dramatically. Web-based technologies provide local government services online in order to make them available to citizens, businesses, and other municipal agencies (Tsai et al., 2009).

Originally, the use of web-based map applications in different municipalities around the world began in many cases with the use of graphical raster maps, which were, in fact, static like paper maps. In many cases, groups of users who wanted to have access to maps but did not have time to learn, install, or maintain GIS desktop software were urged to use the web as a means of sharing geographical data. Finally, both GIS and non-GIS users realized that web mapping is able to combine data from different sources and of different content into one system interface. Access to such a system with the possibility to customize map visualization and spatial data retrieval without acquiring data, creating projects, or learning software, are the advantages of the web-based applications (ESRI, 2008).

Thereafter, rapid growth of computational and data storage capabilities over last years resulted in the possibility of publishing large amount of map data. A significant part of them was published for public use. Thereafter, web sites with multiple user access to GIS resources have been developed in local governments. These changes had tremendous impact on many municipal agencies’ business workflow, government and citizen communication, and many others. Large companies that deal with map production were able to edit their map data interactively via local intranet or the Internet. Citizens were allowed to view a city map on the municipality web site, check parcel tax values via interaction with a map, and contribute to a city map production through evaluation of its accuracy and completeness (Jensen et al., 2005).

Such strong feedback capabilities to map editing, visualization, and analysis stimulated web mapping standards development. Different types of numerous web mapping applications have been published on the Internet both for commercial and non-commercial use (Rinner, 2003). Web mapping application development approach enabled the possibility of wide access to map data to the public and resulted in expansion of business service areas of many companies (Cho, 2005). Yahoo!, Navteq, Microsoft, TeleAtlas – all of these companies expanded their service area to cover demand in access to high-quality maps for various purposes. More recently, the release of Google Maps have demonstrated to web programmers and map users the capabilities of map exploitation and opened literally a floodgate of interest in online mapping via making GoogleMaps API (Application Programming Interface) and KML (Keyhole Markup Language) standard structure open to the public (Turner, 2006).

All of these factors stimulated the development of open-source frameworks for implementing web mapping application as well – OpenLayers, MapServer, GeoServer – to mention the most famous examples. A great number of local government web mapping applications have been implemented based on these platforms over last years. That is particularly true for publishing city map applications of small municipalities in a limited fiscal environment. Along with that, the variety and richness of web mapping service capabilities have been growing incrementally last years in the commercial GIS software sector. Talking specifically about GIS and geodata management, many software packages for automation of publishing maps on the web have been released: ESRI ArcIMS, ESRI ArcGIS Server, Autodesk MapGuide, MapInfo MapXtreme, and Intergraph GeoMedia Web Map – to mention the most famous products. The latter one has been used as a server-side web GIS in the work of Malczewski (1999) whereas most others stuck to ESRI server GIS products (Maroney et al., 2007; Simão et al., 2008; Chang & Park, 2004). The review of these products has been done by Limp (1997) and Culpepper (1998). Since that time, functionality of the products has been changed dramatically.
Nevertheless, with all these solutions being available, it could make the question of which to exploit for a web mapping application development very complicated.

Interest in the programming of map services on the Internet has also been incrementally growing and even had an impact on the standards of GIS software development. The leading GIS software vendors were forced to include web mapping capabilities in their products, to extend programming framework via adding controls for creating and publishing web mapping applications, and finally to provide integration with de facto standards of web maps use – support of KML, OGC capabilities, and many others.

Furthermore, uncountable number of municipalities’ GIS departments’ heads have been farsighted enough to discover the best way of allocating financial and human resources. Successful implementations of web GIS can be observed in dozens of countries. The literature on web GIS topics contains numerous references to tools that have been specifically designed to support local government workflows (Rinner, 1998; Scharl & Tochtermann, 2007; Simão et al., 2008). Numerous web GIS applications have been developed to deliver GIS data, provide citizens with maps, and map analysis functions on the web via the Internet (Batty, 1999; Coleman, 1999; Peng, 1999; Plewe, 1999).

### 3.2 Access to GIS data via the web: concepts

In this research project, the capabilities of GIS software and GIS application development framework are proved for the deployment of such a functionality that local government application users would need. To implement most of the functions that will be mentioned below in desktop GIS software products is relatively simple and a great number of desktop GIS applications exist. However, to deploy such functionality in the web environment is much more complicated, because the interaction between users and a system is performed through a web browser – a very light application, which cannot include by default all programming components required for implementation of GIS functionality.

While distinguishing between static and dynamic web maps, in the scope of the given paper, the use and publishing of static maps will not be considered. This way, to implement a dynamic web mapping application, basically, two approaches exist: client-side or server-side (Peng & Tsou, 2003). In the case of performing all service procedures on the client side, installation of separate executable applications that will connect to the Internet is performed. Alternatively, an installation of a plug-in – a small third-party program – in the web browser is required. For example, Adobe Acrobat plug-in can be installed for displaying portable document format (.pdf) files in the web browser. The client-based approach is worthwhile due to its simplicity; however, the main disadvantage is that a user has to install the given plug-in on the used machine (Kraak, 2001).

Web access to data assumes using a distributed client-server architecture. A web browser is a client software that is capable of sending requests for documents to a web server. A web server, in turn, is a server software that upon a request to a document, process the request from a client and sends the document back (Berner-Lee, 1994). This way, the processing of a request is being performed by the server and the visualization of data is being done on the client side (Figure 4).
In the client-server architecture, a special language of communication is used – Hypertext Transfer Protocol (HTTP) (Berner-Lee, 1994). It exploits the Transmission Control Protocol/Internet Protocol (TCP/IP) for communication between web hosts (Hunt, 1992). All web servers and clients must be enabled to use HTTP in order to communicate between each other (McCauley et al., 1996).

Another approach of a client-side system implementation is related to embedding JavaScript elements and Java interpreters – Java virtual machines – into browsers of end users. Thus, the web functionality can be extended by developing Java programs or including Java script code in the web site pages. The use of Java is becoming more and more popular since it combines the benefits of both client-side and server-side approaches in web mapping development and provides an interoperable environment (i.e., it can be run on multiple OS platforms).

As has been noted by Kraak (2001), server-side systems allow users to query data from the server, and then the server will process the request, retrieve data from a data repository, optionally manipulate the dataset, and finally send it back to a client. Such an approach provides a high level of independence; web applications do not require installation of any additional software on a client side. However, the main issue is related to a server load: to process a great number of queries from users require huge modern hardware and software resources as well as network bandwidth capacity (Table 1). By the load in this case, the number of requests or responds that a side has to process is meant: the higher the number of requests or responds, the higher the load.

<table>
<thead>
<tr>
<th>Architecture type</th>
<th>Client load</th>
<th>Server load</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client-side</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>Server-side</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>Client-server</td>
<td>Can be allocated</td>
<td>Can be allocated</td>
</tr>
</tbody>
</table>

While using mixed solutions (client-server interaction), it becomes possible to interact with geodatasets in a most proper and efficient manner. Such systems allow client users to query map data from the server stored in GIS databases, then the server processes the request, retrieves data from a GIS dataset, optionally performs manipulation of the dataset, and finally sends it back to a client. Web GIS that has been developed by using such methods can use map visualization techniques, and manipulate data, such as download, upload, classify and compare data. From now onwards in the thesis project, combined methods of GIS data web publication will be used (Kraak, 2001).
3.3 Efficacy and issues of web GIS applications

How effective a web mapping application is might be based on its ability to satisfy the needs of its users. Web mapping applications are widely used in numerous local governments for map documents compilation, retrieving address or tax parcel information, and obtaining real estate permissions. For instance, the municipality of Gävle is using ESRI products to accelerate the process of building permit applications. The intranet-based GIS solution delivers accurate and actual information to city agencies staff in a single application in order to replace static hard copy paper maps, which have been used previously to verify real estate property boundaries, land-use information reports, and zoning regulations documents (GIS in Education, 2007).

The application was developed based on the ArcIMS platform, which allows the municipality staff to request building permits and avoid the cumbersome routine procedures. Eddie Larsson, GIS manager at City of Gävle, noted (GIS in Education, 2007) that “with this new application process, the only information applicants need to know is the location and size of the building footprint. From there, the database returns all relevant information, including proximity to neighbors, neighbor permissions requirements, and zoning laws, so that the builder can make the appropriate decisions.” Therefore, during the implementation of a web mapping application of municipal GIS data, several issues should be taken into account: a map should represent correct and up-to-date information and the map should refer to a complete and relevant geographical database. In addition, a web mapping application should include adequate and easy-to-use user interface and effectively manage data user permissions (ESRI, 2007b).

One of the main challenges of during web mapping application development is to discover the most productive and consistent methods of web GIS application maintenance due to the dynamic nature of underlying GIS datasets. User requirements and needs of the web applications might vary greatly. The process of user requirement definitions will not be straightforward. Specialists of system design often notice that users might not know precisely what they want from the application. As previous experience of any computer application showed, effective exploitation of the application scarcely ever can be achieved where designers make suppositions about the needs of end users (Campbell & Masser, 1995).

Since geospatial data are very dynamic, its geometrical and descriptive properties can change dramatically. Therefore, it is important to provide users with actual GIS data via web mapping applications. For these reasons, using unified methods for data updating in a web mapping application is crucial. Further discussion of methods of web mapping application content update is covered later, in Chapter 5 – Web mapping application development: methodology.

3.4 Importance and benefits of web GIS applications

According to previous studies on publishing GIS resources on the web, putting maps and GIS data on the web has several benefits. It is assumed that the popularity of the web-based approach to GIS data sharing and publication will grow, and users will be able to create even more functional web mapping applications. Moreover, in the context of GIS application in local government, the web GIS opens up more opportunities for more people to participate in the public shaping of the local government initiatives rather than traditional meetings and forums (Kingston et al., 2000).

As has been concluded by Arneson & Mitchum (2008), the process of choosing a conceptual platform or business logic for managing all of these GIS data can be complex. However, the recent propagation of web-based tools for managing spatial information has attracted the attention of many experts in GIS. Whereas each approach for GIS data sharing and managing has its own advantages and shortcomings, web-based solutions come with numerous benefits including, but not limited to, easy access to information, the possibility of loading, integrating, evaluating, and exporting of map information via a web interface. All the more so, a web interface provides a tool for customized (i.e., developed for a specific purpose with specific elements and set of tools) geographical information representation that would fit the specific needs of users (Elis et al., 2005; Calder et al., 2008).
Several benefits of use of web GIS applications in local government are as presented below:

- while using web browser for viewing or editing GIS data there is no need to install proprietary software on machines and obtaining licenses
- web-based applications are better suited than traditional desktop GIS since the latter are not equipped for the general public to access, process, and further share spatial information
- while organizing web access to GIS data, administrators have options for setting user permissions to use GIS data and available tools
- developers of web mapping application have access to various settings of data representation: it is possible to customize the appearance of a web mapping application. For instance, geographical coordinates can be hidden in the case of data secrecy, the option of local data copying can be enabled/disabled; editing can be set to be performed remotely, which can solve the issue of the necessity of copying data to a local drive
- the process of multi-user editing and data editing reconciliation are greatly simplified
- using web-based technology allows developers to create interactive, highly customized maps and allows end users to perform different analysis tasks without the need to learn desktop GIS software.

3.5 Development of web mapping applications: issues and recommendations

Based on existing scientific knowledge on publishing GIS documents to the web, it can be noted that web mapping applications are basically all about providing access to map documents or any other GIS database content via a web interface which could be available by using ordinary web browsers. Any web mapping applications should provide the possibility of exploitation information in multi-user mode and should have a graphical user interface (Su et al., 2000).

The process of any web mapping application creation involves map data preparation. Existing GIS map documents should be prepared to be published on the web; the user interface should be designed properly, so end users’ needs and level of skills could be addressed adequately (Rees et al., 2001). During implementation of web mapping applications numerous challenges could occur. Several authors (Maroney et al., 2007) emphasized the importance of data quality, which makes the difference in exploitation of those applications. Data errors might not be noticed via GIS spreadsheets, but may be easily observed when viewing the web mapping application, particularly while organizing public access to GIS information.

Another crucial aspect of web GIS application preparation and publication to the Internet is that it requires efficient communication between different specialists within the organization. As has been shown in similar projects (Maroney et al., 2007), such categories of users as database administrators (DBA), web administrators and developers, computer security experts, cartographers, and GIS software experts might be involved in the process of web GIS applications development.

While implementing web GIS applications particularly by using ESRI GIS products, there is a need to keep up-to-date with the latest versions of the software, service packs, and patches. Moreover, it is crucial to follow the compatibility between software products of different versions since the updating of a certain component of a system to a newer version, could lead to the crash of the whole system (Maroney et al., 2007). However, the understanding of that can come already after updating of a system element and that can put the whole system work in danger.

Moreover, GIS and computer knowledge can vary greatly among end users of web GIS applications. While designing a mapping application, which is supposed to be fully web-based, several recommendations could be extracted from Svanberg & Winkvist, 2008. Their observations suggest that such an application should be designed in a very clear way. Designers should make sure users understand that the web browser is just a tool for accessing the true web system and is not part of the GIS. What is amazing is that during their research, they performed a user survey on the use of web-based GIS and several participants during the test at least once searched the context menus of the web browser in order to seek aid in reaching their goal in interaction with map features.
Moreover, while designing a web-based system, one has to keep in mind that multiple web browsers can be used by end users and they might be not compatible with each other. Furthermore, different implementations of the system could be needed. Indeed, it would not make sense to make an application web-based in order to get a wider audience and then to limit the number of users to only one web browser (Rees et al., 2001).
4 Web GIS development in the municipality of Gävle: case study

To date, the need to implement more advanced tasks and to provide better performance in web mapping applications has occurred in the municipality of Gävle. A framework for implementing a web mapping application that could satisfy extended user needs is provided by the ArcGIS Server software from ESRI, which has been chosen for the research project due to several reasons.

First and foremost, the municipality of Gävle has the Small Enterprise support license (so called ESRI ArcGIS Community License) and owns licenses for a majority of ESRI ArcGIS Desktop family products. This means that the municipality can install any ArcGIS product to use within the municipality. Second, according to recent employment of the ArcGIS Server in academia, it can be highly recommended as a product, which can incorporate organization’s workflow environment and perform required tasks efficiently (Fan et al., 2005).

Furthermore, ArcGIS Server could serve as the development platform for web GIS implementation and customization. Even though ArcIMS would enable the same functionality, it would require extra programming (Simão et al., 2008). In addition, the municipality of Gävle has been using the ESRI GIS software over last decade and staff is mostly familiar with this software logic. Thus, ESRI ArcGIS Server 9.3 has been chosen as the map server for serving GIS mapping functionality on the web.

ArcGIS Server can be treated as the GIS application server. It enables the implementation of GIS business logic in an IT standard-based server environment within an organization. Previously, such a business logic was available only in GIS desktop solutions. Nowadays, however, ArcGIS Server allows users throughout the local government to access GIS capabilities via a single shared system. Thus, the municipality could build one solution and deploy it to multiple users (ESRI, 2006). ArcGIS Server is a flexible and scalable system, which solves a wide range of tasks: sharing GIS data, publishing GIS resources on the web, or distributing GIS data among users. The structure of the product allows administrators to distribute hardware and system resources wisely and have thorough methods of system administration and integration (ESRI, 2003).

To summarize, ESRI ArcGIS Server was designed to provide a framework for publishing and sharing GIS resources. Users are able to create out-of-the-box template-based web mapping applications by using a wizard in a step-by-step manner. This has allowed developers to leave all technical processes behind the scenes and thus simplify the user-system interaction since no expertise in programming and system administration is required. Therefore, the task of developing a web mapping application with advanced GIS tools could be solved relatively easily nowadays; however, just about 10 years ago, that would have required extensive programming skills and expertise in GIS and web site administration (Giannotti & Pedreschi, 2008).

4.1 Demand on new web GIS platform in the municipality of Gävle

With the growth of volume of data at such a fast pace there was an increased need for advanced GIS applications in the GI-department. City residents, surveying companies, and staff of other different city agencies used to spend a lot of time to obtain information from the GI-department. With a great amount of time being spent researching properties, it could be harder for the specialists to focus on their main job tasks.

Thus, in the last years, GIS staff proposed to create several web sites that would allow public users to search various required information. This would help to reduce the number of people requesting information immediately in the office and through e-communication and would allow anyone to search for data any time. More detailed information has been covered in section 2.2 – Types of data and main applications of GIS in the municipality of Gävle.

As was introduced previously, originally, the ESRI ArcIMS (Internet Map Server) software was chosen. However, when data volumes and user needs expanded, it was realized that the ArcIMS-based application has several limitations. There are several very thorough technical resources, which include the description of ArcIMS-based and ArcGIS Server-based web mapping applications, for example,
ESRI (2007a). In short, it is relatively difficult to administer the ArcIMS-based system since it consists of several independent third-party components (Apache/Microsoft Internet Information Services (IIS) as a web server, Tomcat/ServletExec as a servlet container, Java Server Pages (JSP) as servlet, and Java Virtual Machine (VM)). Even though in ArcIMS 9.2/9.3 numerous enhancements including, but not limited to, better performance, better data security, and new development platform support have been introduced, analogous ArcGIS Server 9.3 based web mapping applications provide significantly better performance and capabilities.

Thereafter, when ESRI ArcGIS Server 9.3 was announced, the decision to migrate to a more powerful, easy-to-administer, flexible, and scalable web-based application has been made in the municipality. The advantage of the given software is that the same web application is able to provide both internal access to government employees (via a local network, an intranet) and external access to all authorized external users (via the global network, the Internet). Description and benefits of the ArcGIS Server have been listed in the beginning of Chapter 4. To conclude, the task of the staff of the GI-department is to evaluate how effectively ArcGIS Server platform can be used to create fast and functional mapping applications that could be used instead of existing ArcIMS-based web application.

4.2 Requirements for the target web mapping application: system analysis

Before starting a web site many questions have to be answered. What data or layers of information will users require while working with the designed web application and what functions do these users expect from the application? Who were the clients that would be accessing this web site? How often would they access the web site? What would these clients be able to see when accessing the web site? What are the means clients could use to view the web site? By answering these questions, the GIS staff will be able to develop a web site that could address needs of the target audience. It is also important to provide users with the opportunity to use different client software (particularly, thin clients), for example, to view the website.

Thus, the main idea of any web mapping application development is that the application should satisfy the needs of its users. The aim of any map application is to provide a means for people to explore the map and properties of its features. Many authors mention the necessity of fast drawing of a web map since users expect very dynamic interaction with a web mapping application (Kraak, 1991). As for other requirements, the ability to quickly visualize layers and the option of switching on/off layers should be mentioned. If applicable, an application could leverage from having an option of adding local data or data from the Internet as map layers (Maroney et al., 2007). Previous developments in GIS server linked with a web server resulted in providing users with some limited GIS functionality on the web (Conquest & Speer, 1996). Later the distributed system approach was used in order to distribute data and GIS processing elements from the GIS server to web clients (Jiang, 2001). These techniques demonstrated the most efficient use and appeared to be very scalable.

In very advanced web mapping applications, users could also change the symbology of map layers and create map graphics interactively. Moreover, the representation of the results of buffer tools (selection of objects of a specified layer which fall within the buffer area) as a table of results with features attributes might be required. Thereafter, this table could be printed or exported as a spreadsheet file for further analysis (Maroney et al., 2007). The necessity of interactive compilation of a new map document by a user in a web mapping application by adding local layers from the host or layers from other servers is mentioned in scientific articles as well. Like other tools, the use of map algebra functions, overlays, and vector-raster conversion tools could be mentioned (McCauley et al., 1996). However, the development of these advanced map manipulation tools is relatively hard and will be left for future research.

When loading a start page from a web mapping application it could be also worthwhile to allow users to choose the quality of images that will be drawn in the application. It could allow users with poor broadband connection to have almost the same level of performance as one would have while working with the best quality of images with a wide broadband connection (McCauley et al., 1996). The techniques, which could increase the performance of map data serving to the client and reduce the server computation time upon retrieving maps, will be covered in detail in Chapter 5 – Web mapping application development: methodology.
Typical web mapping applications should include such map elements as a legend, north arrow, scale bar, navigation tools, and title. Besides standard tools set, tools for viewing object coordinates and exporting maps to a PDF-file or an image for future printing or electronic distribution are usually required. Access to a web mapping application, if it is a secured application, should be restricted by using an authentication dialog box. User permissions will define the possibility of user access to map features and map export functions (Simão et al., 2008; ESRI, 2008).

An additional requirement is the option of linking map feature data with external database rows, report files, or multimedia data via hyperlinks. In a web mapping application, it is possible to have hyperlinks, which could include not only text, but also multimedia forms, i.e. photo and video documents. Using rich text formatting and HTML for hyperlinks can allow developers to customize the web site appearance while embedding a map element into its design. Even though web mapping application engines will interact with some advanced programming framework like Active Server Pages (ASP) technology from Microsoft and JavaScript, some aspects of HTML design (changing colors, fonts, adding web forms, and so forth) will most likely still be required. Support of styled layer descriptors (SLD) format should be provided as well in order to provide the interoperability with OGC standards.

Web mapping applications should provide access to the help system as well, where users could read about how to use map interaction tools and can obtain contact information for feedback. A help system is important since its content can be updated immediately after finding errors or shortcomings; thus, users can give a feedback to developers and finally contribute to the web application implementation. Finally, the map application viewer should be capable of being integrated into existing web sites (McCauley et al., 1996).

Important issues identified from related works (McCauley et al., 1996; Peng, 2001), include the need for empirical user studies. This implies that web mapping applications should include a graphical user interface (GUI) that could enable users to manipulate map information and map layers over the Internet. Users should be able to locate map data they need, get access to the required map image in an operative manner, and finally upload or view the map image. Furthermore, the process of user-system interaction should be as simple as possible, so even the novice user would be able to interact with a web mapping application effectively.

Several authors (McCauley et al., 1996; Tsai et al., 2009; Peng, 2001) paid attention to the security question in web mapping applications as well. Security holes that could give malicious users access to a web server or a host operating system should be eliminated. Any web mapping application is required to be modified and it is not recommended to use web application via the Internet without security related modifications. However, a more detailed description of security issues is not in the scope of this thesis report. From a technical perspective, regarding a specific web mapping application for local government function requirements, several main needs could be mentioned (Peng, 2001):

- the web mapping application should be Internet-based and be neutral from the perspective of operating system interoperability to allow wide access. That is, client-server architecture should be adopted and a user interface should be used that does not restrict users from the use of the application due to the computer platform used
- a system should support a distributed database management system model in order to allow users to visualize data from different sources – local and remote
- a HTML-based graphic user interface with a map should be provided in order to allow users to interact with map features in the web browser; users should be able to perform a spatial search and query, redline editing, and map image exporting such as the .pdf, raster images, and vector images, if applicable
- the system has to be compatible with International Standard Organization (ISO) and if possible with OGC standards since it could help to integrate other standardized GIS data resources.
The proposed web application system objectives can be outlined as:

- To allow users to create maps over their area of interest via the Internet
- To allow users to create short reports on records of map data
- To allow users to perform a search of address data and real estate information on the web site
- To facilitate integration of existing base maps into one web mapping application
- Complement existing GI-department mapping and analysis web-tools, if applicable
- Permit the public to access the existing document management system resources and other municipality web-based systems, if applicable
- Allow both external and internal users to retrieve the information and analysis tools from the web.

4.3 ArcGIS Server as a platform for local government web GIS applications

In Figure 5, one can see the schema of ArcGIS Server installation in the municipality of Gävle. It includes several machines that are united in a local network. Such an installation is called a distributed installation according to ESRI documentation terminology (ESRI, 2008).

The GI-department at the municipality of Gävle strives to develop an application, which would address all standards of enterprise level system development. It should be possible to interact with a web application in a standard web browser with no plug-in for the end user environment; an open standard architecture for the development and further customization of the application should be used. This implies that the developers at the municipality should be able to tailor the web application by using an open development framework, for instance, Microsoft .NET or Sun Microsystems Java.
Finally, an application should be able to serve both raster and vector maps to end users over the Internet.

Most of the predefined above requirements could be satisfied by using the ArcGIS Server platform. Moreover, this product addresses several other requirements of the GI-department that were also stressed. This product is compatible with all other ESRI GIS software products that are being used in the municipality. Finally, ArcGIS Server is capable of integrating with DBMS Microsoft SQL Server and hence, it will be possible to integrate existing database information in any web solution.
5 Web mapping application development: methodology

As one of the current tasks, to develop a new web mapping application similar to the cX-Länskartan application (see Chapter 2 to learn more about the applications in the municipality of Gävle) is required. This application should include the map for the whole area of the Gävleborg county and will be used for public access. However, to determine the best model of a web mapping application design, development, and customization, the limited area that includes merely the map of the municipality of Gävle has been used. In other words, a prototype of the production application should be created by using data for a smaller geographical area in order to test the planned workflow schema and demonstrate the system capabilities (Huxhold, 1991). As several authors have concluded, the most adequate way to design and implement a GIS application for municipal needs is to develop a prototype of mapping application and then perform incremental increases of the power of GIS data, capabilities, and functionality (Huxhold, 1991; Tikunov, 2004). The advantage of this method is in the ability of flexible phased development and immediate start of consumption of web GIS application for different purposes.

This way, it seems to be reasonable to develop a prototype of a web mapping application and then to make sure that system requirements and task requirements are understood in a proper way. Finally, any corrections and changes according to the feedback from users can be made to reflect their requirements of the web mapping application’s content, user interface, and functionality. Indeed, implementing projects first in a test environment instead of a production environment and surveying the users’ needs could tremendously help GIS developers to retrieve accurate information about the existing data, data accuracy, skills of staff, and user requirements and to allow users to become familiar with the application (Dragićević, 2004). Incremental web GIS application development is very useful because of the significant number of system elements and the necessity of testing the application in a production environment. Moreover, actual user needs can be discovered only in the process of using the web mapping application rather than by system design expertise or through single user survey (Bowen, 2005).

Obviously, the new application being developed is based on ArcGIS Server platform and should provide at least a better performance than the analogous existing ArcIMS-based application, which has the benefit of fairly fast map rendering. Thus, during such a project carrying it out is required to provide the opportunity for the GI-department to test the application performance, to verify user acceptance, demonstrate its ability to improve public awareness, and to provide map interaction tools. Such a practice is also very common among many system designers: they use pilot projects to produce results quickly, and to analyze user needs and technological limitations (Heywood, 2006).

Bernhardsen (2002) outlined the conclusions of a pilot project undertaken in Oslo by the local government. Applying a desktop GIS project concept to a web environment, a developed web mapping application could show how well it meets the needs of municipality and external users. Such a pilot project could entail a small-scale implementation of the web mapping application where the tools, interface, and the performance of a full-scale production application could be tested in advance. During a prototype application development, a better basis for the possible choice of system(s) should be provided; several production methods should be tested; finally, system faults and bottlenecks should be identified.

Developing a web mapping application in ArcGIS Server environment includes several steps. Firstly, it is required to create an ArcGIS Desktop map document. This document will provide a map that will be put in a web application. After creating a map, it will be published as a map service and then as a web mapping application. Thereafter, the web mapping application (WMA) will be customized to fit user needs in the best way (Figure 6).
Chapter 5 will discuss the methods of implementing this project workflow and Chapter 6 will show the results of the project development according to the workflow guidelines in the figure above (Figure 6). This schema is based on iterations, i.e. after creating a web mapping application, its performance should be evaluated, and if it is not satisfactory, one goes to the previous step. That means that even after going through the stage of creating a web mapping application it can still be required to go several steps back to optimize the map data redrawing performance in the map document or to apply another method of map service caching in order to provide the required performance.

### 5.1 Map document preparation

Firstly, to create an ArcGIS Desktop map document, a conceptual model of a map document is needed. One should know which layers and which attributes are required to be shown on the map. This model is independent of the ArcGIS Desktop capabilities and it represents the user’s view of the map and its elements.

After creating the map document, it is beneficial to consider the organization of access to GIS data for future publication as well. Since ArcGIS Server is a distributed system, it requires proper communication between its components. This means that all system elements should have appropriate permissions and access to GIS documents, resources, and services. Moreover, as was mentioned previously, the process of web mapping application implementation would require the communication between different department staff. ArcGIS Server GIS Server would need to have access to database tables, so the database administrator should make sure that the database would be accessible. In the case of the municipality of Gävle, a distributed system installation has been performed. This means that security issues should also be taken into account since operating system firewalls can block the machine’s ports, which might be used to communicate between ArcGIS Server system elements. System administrators and IT-department should perform the setting of operating system properties.

Optimization of a map document for further publication is the next step in the project. As acknowledged by Batcheller et al. (2007), a system will only work as fast as its slowest component. Indeed, the web application’s map will not be rendered faster in the web browser than it is rendered in the ArcGIS Desktop application since this map is based on the same map document. Thus, various performance evaluation methods should be applied in order to seek the best possible performance of map redrawing.
The optimization of a map document drawing performance can be done by using a special utility from ESRI – Mxdperfstat. ArcGIS 9.3.1, a release that is being shipped at the moment of writing, includes most of this utility functionality as a toolbar. However, at the moment this free utility is the most appropriate ArcGIS-compatible solution for map rendering performance evaluation. This software from ESRI Testing Service is widely used to help diagnose typical performance problems with ArcGIS map documents such as (ESRI, 2009c):

- Inefficient scale dependency options use
- Slow symbology rendering
- Large number of features
- Using projection on-the-fly
- Potential database tuning needs.

5.2 Publication of the map document as a map service

The publication of a map document as a map service is a process that can make the map document available to the different components of the ArcGIS Server platform. The published map service can be consumed directly by different desktop clients from both ESRI and third-party applications. The publication of the map document as a map service will allow developers to create a web mapping application that will use this map resource content. Evaluation of various map service parameters should be done in order to provide the best possible performance of server-service-client interaction.

5.3 Caching the map service content

After creating a map service, it should be cached at scales that users will look at most often. Caching techniques are very important in the context of web mapping applications, because they provide the opportunity of fast visualization of a map for a large geographical area with thousands of features and sometimes even provide a better performance than the desktop GIS applications.

In the ESRI ArcGIS Server 9.3 platform, several map caching techniques for GIS map services are provided. A cached map service is an ordinary map service that has been enhanced to redraw maps in the web browser very quickly by using a cache of static images that have been precompiled beforehand (ESRI, 2009b). Advantages of using of caching techniques in a web mapping application are particularly striking when manipulating large GIS datasets as in this research project where the map of the whole municipality should be visualized.

The use of caching techniques in ArcGIS Server environment has not been extensively covered in scientific papers, instead mostly in technical papers, for example, in ESRI (2008). However, caching as a technique for optimization of map service data visualization has been envisaged in numerous articles (Wei et al., 1999; Tu et al., 2001). Their research has shown the efficacy of the caching techniques for spatial data transmission via the Internet.

The basic idea of caching is relatively simple. The whole map image is divided into multiple parts, which are referred to as tiles (Figure 8). When a user will view a certain map area, the GIS server will not submit the whole map image to a client, but will instead send the client tiles of the very region, which the user will view at the moment. The received map tiles will be stored locally on the client machine for future reuse (Wei et al., 1999).

This way, the map cache is a set of image files that are stored in ArcGIS Server cache directory. The map service cache contains image tiles of an extent of a map document at the predefined specific scale levels. Map caching improves the performance of a map service data rendering greatly and reduces the load on the GIS Server. Thanks to caching, clients can retrieve tiles directly from the cache rather than having the server dynamically generate the map images (Figure 7).
In this research project, the job of caching the map of the municipality of Gävle was relatively straightforward and was completed in a few hours. However, a larger caching job (the caching of the entire Gävleborg county at several large scales, for instance) could take several days to complete (of course will depend on allocated memory resources). Hence, it would be a good idea to have a general strategy of caching, so one will not waste time while building a cache merely in order to discover that the cache image quality and scale range do not fit needs of the web application end users (ESRI, 2008).

Moreover, one should keep in mind that the process of cache generation could be very time- and CPU-intensive. Previously performed optimization of the ArcGIS Desktop .mxd map document could not only increase the speed of the vector data rendering in a web mapping application that will be based on a non-cached map service, but could also decrease the time required for cache generation.

Another task is to choose the tiling schema format. To create a map service cache, several cache tile schemas can be used (ESRI, 2009b):

- **ArcGIS Online.** It is the standard schema that is used in ESRI ArcGIS Online web mapping applications. In this schema, the PNG image format is used with the 512x512 pixels image tile size.

- **Google Earth/Virtual Earth.** To use this schema it is required to use the Universal Transverse Mercator (UTM) projection and a NAD27 datum with WGS84 parameters for GIS data that are supposed to be published. It is the standard schema that is used in Google and Microsoft mapping applications. In this schema, the PNG image format is used with the 256x256 pixels image tile size.

- **Custom.** While using this schema, ArcGIS Server users are provided with the opportunity to create a custom configuration file that will define required settings. In this schema, the custom image format and tile size can be set.

- **A schema that has been used in other existing map services might be reused for other services as well.**

One should keep in mind that if the decision to use Google Earth/Virtual Earth tiling schema should be made, then all data should be represented by using WGS 1984 Web Mercator projection. There is a special projection for building online maps in ArcGIS Desktop that is referred to as WGS
1984 Web Mercator. Data that are stored in different projections can be reprojected to WGS 1984 Web Mercator Auxiliary Sphere projection. Moreover, if one will make a decision to use Google Maps tiling schema, appropriate scale dependencies should be set in the ArcMap document. This means that ArcMap users should operate in Google Maps scale environment while preparing an ArcMap map document. Google Maps scales can be loaded to ArcMap’s user interface thus making a set of Google Maps scales available to ArcGIS Desktop users (ESRI, 2009b).

While working with Google Maps scale range, it is possible to reset to standard scale range in ArcMap application. This implies that there is a need to design a map to properly look at those scales at which the map document will be cached later. Therefore, if it is not planned to cache the map document at all map specified scales, then it is required to design a map for use at those map scales, which a web mapping application should support. There is no need to do any map design for the other map scales that will not be used in a web mapping application, because end users will not use them (ESRI, 2009b).

Choosing a color schema of cache images is another trade-off process since each of them has its own benefits and disadvantages. Below is the brief description of each of the available formats for image tiles (ESRI, 2009b):

- **PNG8**. It is a lossless, 8-bit color, image format that uses an indexed color palette. Its transparency parameters are stored in the color index palette. PNG8 provides excellent web browser support.
- **PNG24**. It is a lossless, three-channel image format that supports large color variations (up to 16 million colors), but it has limited support for transparency. The transparency value is stored in the image header. Furthermore, versions of Microsoft Internet Explorer lower than version 7 do not support this type of transparency.
- **PNG32**. It is a lossless, four-channel image format that supports large color variations (up to 16 million colors) and transparency settings. PNG32 provides excellent web browser support.
- **JPEG**. It is a lossy, three-channel image format that supports large color variations (up to 16 million colors), but does not support any transparency capabilities. JPEG provides excellent web browser support.

ArcGIS Server also provides the possibility to choose a type of map caching: fused or multi-layer. Multi-layer caching gives end users the opportunity to switch cached layers on/off after performing caching procedure because the cache image tiles are created for each layer individually. Fused caching performs caching of the whole map image as one layer. As has been acknowledged by ESRI (2009d), the fused caching technique provides the best level of performance. However, in the final web mapping application users will have no possibility to toggle layers on/off.

### 5.4 Creating a web mapping application

After finishing preparing a map service and map service caching, it is possible to create a web mapping application. The web mapping application can be created by using a template from ArcGIS Server Manager – a fully web-based application for administering and using ArcGIS Server system or programmatically by using tools from Microsoft Visual Studio and ArcGIS Server Web ADF. At this stage, a standard template of web mapping application should be used in ArcGIS Server Manager.

### 5.5 Evaluation of performance of the web mapping application

To assess the performance of a developed web mapping application, different methods could be used. The adoption of ArcGIS Server as the main platform for web GIS application development has been made in order to increase the speed of map rendering compared to the existing ArcIMS-based application. Hence, it seems reasonable to compare it relative to the performance of previous ArcIMS-based applications, that is, to perform benchmarking (Batcheller et al., 2007). Web mapping application performance should also be performed in different web browsers. First, performance testing should be done in a development environment, and then after web application deployment on a production server on production user load.
5.6 Customization of the web mapping application

In numerous projects that have been done for local governments as well as for other sector needs, standard out-of-the-box templates for web mapping functionality do not provide several tools that are highly required. In this research, the default ArcGIS Server web mapping application should be extended by compiling or developing custom tools to interact with the map. During the project application customization, one should keep in mind that those users who are unfamiliar with GIS technology might experience difficulties in using the map. Despite the tooltips shown when a mouse cursor hovers over a certain button, it could still not be obvious to users how they should manipulate the map, to control the map layers visibility, and to use other tools for measuring distance over the map, to mention just a few examples.

For developing and/or compiling of existing code and system tools, Microsoft Visual Studio 2008 has been used. This programming framework is widely adopted as the de-facto standard in desktop as well as web application development. Visual Studio tools are widely accepted by the ESRI GIS community as well. ESRI provides the integration of Visual Studio and a special web GIS development framework called ArcGIS Server Web ADF. This environment can be used to perform a customization of the project web mapping application.
Figure 8. Representation of map tiles in different scales. The maps show increasingly larger scales, which represents objects with increasing level of detail.
6 Web mapping application development: results

6.1 Map document preparation

While creating a conceptual model of a map document, ArcGIS Diagrammer – a free application from ESRI has been used to model a logical geodatabase that will be a source of map document layers. For identification purposes, several layers – real estate data, address data, and data on important objects – have been added to a separate map document. Access to a database with different information on city objects has been provided as well. All required ArcGIS Server system user accounts permissions have been set properly in order to provide access to GIS data for different system components. While experimenting with different parameters and settings in the current project system and analyzing ESRI GIS expert recommendations by using Mxdperfstat utility from ESRI Testing Service, the list of map document requirements have been compiled. Some of these results have been acknowledged by previous researches, others are newly introduced in the current work. These requirements are presented below and could be used as a guideline for any future research on map document optimization for its publication in an ArcGIS Server web mapping application:

1) It is required to increase the speed of layer rendering as much as possible in an ArcGIS Desktop application, since the web application map will not render faster than in ArcGIS Desktop. This can be done by decreasing the number of features and simplifying the geometry of features.

2) It is required to minimize the number of layers in the map document content and to use definition queries wherever possible. In the given project, multiple definition queries are used. The same feature class can be referred to by several layers in the table of contents by using definition queries. This method proved to be better than the use of attributes for displaying different classes of layers with varying symbology.

3) It is not recommended to use Join or Relate functions in ArcGIS Desktop since that greatly decreases the performance of map feature retrieval. Wherever it is possible, required fields from joined or related tables should be transferred to the feature class attribute table that will be used for further identification.

4) It is required to use a special set of cartographic symbols that is called ESRI_Optimized and is available in ArcMap application for map layers symbology. It is required avoiding using complex symbology and labels for vector map objects as well. To create and use attribute indexes for fields that are used in the definition queries or for fields based on which the search will be performed is highly recommended.

5) To create and use spatial indexes for all layers is recommended. It could be also a good idea to customize values suggested by the system by default (the performance of spatial operations with map features can be expected to be significantly improved for feature classes with associated spatial indices (Batcheller et al., 2007)). Default database indices might be specified inadequately, or sometimes even be absent. However, as my tests confirmed, spatial indices could not influence the performance of attribute querying, but for spatial operations indices play a crucial role, particularly after transferring project data to the ArcSDE geodatabase classes, which are allowed to have up to three levels of index grids.

6) It is recommended to have all data on the map in the same projection as the map document data frame (to prevent on-the-fly transformation that will consume some system resources).

7) It is necessary to have all data on the map stored in the same source (i.e., the same database). Having map document layers from different databases will influence data drawing performance negatively. If using a personal/file geodatabase, then relative links to the source files should be used. If the ArcSDE geodatabase is used, then it is required to provide access to the .sde connection file (in particular, to the one created during the ArcGIS Server installation SOC account). The file .sde is
stored in C:\Documents and Settings\<user_name>\Application Data\ESRI\ArcCatalog). Nowadays the three-tier architecture of ArcSDE is used for GIS data access in the municipality of Gävle. However, from June 2009 the shift to use direct connection as an enterprise standard of the access to ArcSDE spatial databases will be performed.

During this research project, ideas from Batcheller et al. (2007) have been confirmed. Indeed, direct connection architecture is in general more efficient, because the demand on the server is reduced, better failover support can be provided, and finally there is no need to administer any ArcSDE server processes. As my performance tests have shown, the use of the ArcSDE repository for GIS data with direct connection to a DBMS provides the best performance in rendering attribute and spatial querying compared to use of the ArcSDE with the three-tier architecture.

8) Using a reference scale for any layers in the map document is not recommended since that could influence performance negatively.

9) Using ArcSDE geodatabases instead of shapefiles or single-use geodatabases is required wherever it is possible. That is particularly true when deploying a network solution. With ArcSDE that functions as a “data filter,” only the information relevant to the particular query operation will be retrieved from the network data source repository. File-based approaches, in turn, require that the entire object class be loaded to the client side and processed later. This procedure will be time-intensive particular for huge datasets and will lead to poor performance (Batcheller et al., 2007).

Some other minor recommendations are outlined below as well:

- It is required to use only annotations for object labeling. In the project, labels have been allocated on the map by using an intelligent Maplex engine in the ArcMap application and then converted the labels to annotations stored in a geodatabase
- It is required to use scale dependencies to decrease redrawing time
- It is a good idea to use the same scale dependencies for labels as used for layers
- To provide the best possible performance, scale values have been defined with geometrical progression.

During this project phase, two ArcGIS Desktop map documents (.mxd format) have been created. One includes only those layers which will be used for queries (query map); another includes all layers that will be cached later (reference map). The reference .mxd document will be used for caching, and the query map document will be used for map service data retrieval. The ArcGIS Desktop map document for cX-Länskartan of the municipality of Gävle area includes multiple layers, namely, Municipality borders, Land use data, Hydrographic data, Water objects, Communication data, Buildings, Address data, Real estate data, Outstanding objects, and several other layers.

Data were stored originally in a file geodatabase since it provides a relatively high performance compared to a personal database (MS Access format, .mdb), but were then converted to Microsoft SQL Server format to provide the best performance. Map data are projected by using projection RT38_25_gon_V. After creating the map documents, they were published on ArcGIS Server as two map services.

6.2 Publication of the map document as a map service

This process is very simple and straightforward and has been done by using the ArcCatalog application as well as by using ArcGIS Server Manager web application tools. Two map services have been published on the GIS Server and appropriate properties of the map services have been set up.

6.3 Caching the map service content

In this study, fused map cache was created by using PNG8 image format, 256x256 pixels size, 96 dpi. Map cache in eleven cache scales based on Google Earth tiling schema were created during the research project. It took around 10 hours to produce a cache in these scales for the whole area of the
municipality of Gävle within the defined scales. Summary table of the cache generation process is below (Table 2).

After creating the map cache, an important question of map cache update is raised. Map cache can be updated automatically or manually. The frequency of map cache update will strongly depend upon how often underlying data are being changed. In web applications of emergency agencies such as police, ambulance, or fire crews, the presence of up-to-date map data is crucial and, therefore, update of map cache might be performed on a day-by-day basis. In the case of the given project, however, as has been preliminarily decided by me, map cache update has much less significance and can be updated on a week-by-week basis.

Table 2. Cache generation summary.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image format</td>
<td>PNG8</td>
</tr>
<tr>
<td>Image tile size, pixels</td>
<td>256x256</td>
</tr>
<tr>
<td>Resolution, dots per inch (DPI)</td>
<td>96</td>
</tr>
<tr>
<td>Cache scales</td>
<td>1:1,128, 1:2,256, 1:4,513, 1:9,027, 1:18,055, 1:36,111, 1:72,223, 1:144,447, 1:288,895, 1:577,990, 1:1,155,581</td>
</tr>
<tr>
<td>Time of creation, hours</td>
<td>Around 10.5</td>
</tr>
<tr>
<td>Number of folders</td>
<td>Around 2600</td>
</tr>
<tr>
<td>Number of files</td>
<td>Around 3,000,000</td>
</tr>
<tr>
<td>Total weight of cache, GB</td>
<td>Around 11</td>
</tr>
</tbody>
</table>

The creation and update of the map cache for the prepared map service in the 1:1,128 map scale took a very long time and required significant disk capacity, and that was merely the data for the municipality of Gävle. The whole Gävleborg county is around 11 times bigger than the area of the municipality of Gävle. Thus, it could be concluded that the creation of map cache in this scale range for the whole county would require several days (will depend on allocated server resources) and several dozens of gigabytes of disk storage (will depend upon chosen image type and resolution). This type of resource allocation might be considered unacceptable. Hence, it has been decided that to use the map cache in 1:1,128 is not reasonable and instead the vector data of a dynamic map service will be used. That is, while moving over the map in any scale, end users will see the cached map service image tiles, but when they zoom to 1:1,128 map scale, the vector reference map will be rendered and no cache images will be retrieved since cache tiles have not been created.

Nevertheless, updating the map cache will be required anyway. However, the map of the municipality of Gävle represents a relatively large area. It is made up of multiple layers, each of which represents detailed datasets and its cache update is a very resource- and time-consuming process. This way, it would be required to answer the question of which map areas and which data layers of the developed web mapping application should be updated. Only a few map layers will be updated over time. Since the fused cache is used, the update of the whole map cache image tiles would be required. To prevent the update of the cache of the whole area of the municipality of Gävle, the cache should be updated based on the specified area – the extent of buildings layers where object geometry can be changed (Figure 10).

The test of caching defined areas has shown that when seeking the best performance during cache updates by feature class, generalization of features in this class is required. For this purpose, map
objects have been dissolved to several objects. Automatic updating of a map cache for a specified layer has been realized by using a Python script, which could be scheduled to run by using Windows operating system tools. This has been implemented in ArcGIS environment that proved to have very powerful capabilities of this procedure automation. The used Python script can be found in Appendix B.

Another solution to this issue has been implemented by analyzing geographical database versions. It can be realized via techniques of versioning in ArcSDE geodatabases. In this model, differences between objects in the original (cached) feature class and the production feature class are first found, then based on these the envelope objects are created and lastly, based on these envelopes the cache update is performed. The created geoprocessing model and its Python script are found in Appendix B.

6.4 Creating a web mapping application

After finishing preparing a map service, it is possible to create a web mapping application. This has been performed by using web application ArcGIS Server Manager. The created ArcGIS Server web mapping application for the geographical area of the municipality of Gävle included the tools and components illustrated in Figure 11 and described in Table 3.
Table 3. Out-of-the-box web mapping application tools.

<table>
<thead>
<tr>
<th>Application element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Navigation tools</td>
<td>Tool for manipulating the map: Zoom In/Out, Pan, Full extent, Previous extent, Next extent, Identify, Measure, Overview map.</td>
</tr>
<tr>
<td>2. Map element</td>
<td>Map of GIS document that has been published by using ArcGIS Server</td>
</tr>
<tr>
<td>3. Table of contents</td>
<td>List of map layers and their symbology</td>
</tr>
<tr>
<td>4. Name of the web application</td>
<td>Name of the web mapping application that end users will be able to see in their web browsers</td>
</tr>
<tr>
<td>5. Results</td>
<td>List of attributes of selected map features</td>
</tr>
<tr>
<td>6. Scale bar</td>
<td>Scale of the map in predefined map units</td>
</tr>
<tr>
<td>7. HTML hyperlinks</td>
<td>Hyperlinks to predefined external/internal web pages</td>
</tr>
<tr>
<td>8. Navigation bar</td>
<td>Tool for navigating over the map</td>
</tr>
</tbody>
</table>

After creating a standard out-of-the-box web mapping application, several tasks of the web mapping application have been added by using standard tools in ArcGIS Server. All of those operations can be performed in the ArcGIS Server Manager web application (Table 4).

Table 4. Out-of-the-box added tasks to the web mapping application.

<table>
<thead>
<tr>
<th>Added tasks</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Print task</td>
<td>The task allows end users to print a map to an image in a new web page. This task allows end users to include results of the search, results of map features identification, and map layer legends to a print page.</td>
<td>If several map services are used, the Internet connection to map services should be used during the creation a web application (but not local connections to services). Print task allows end users to print a map image that could include data from dynamic vector map services and cached services.</td>
</tr>
<tr>
<td>Search address task</td>
<td>This task allows end users to perform a search of addresses of objects based on certain field(s) values. Once the search has been done, the searched objects will be highlighted on the map. Results of the search can be printed out to a separate web page with or without the map and legend.</td>
<td></td>
</tr>
</tbody>
</table>
The task allows end users to perform a search of real estate database objects based on certain field(s) values. Once the search has been done, the searched real estate objects will be highlighted on the map and their attributes will be available in a table format. The possibility of non-case sensitive searches in attribute fields has been implemented as well.

Results of the real estate objects search can be printed out to a separate web page with or without the map and/or legend. The search in ArcGIS Server Query Attributes task is case sensitive. However, a workaround has been used. The custom WHERE expression has been implemented that transfers both the entered text string and the stored in a database row value to lower case and performs comparison based on a lower case.

Here is the used SQL expression:

```sql
lower(NAMN) like lower('%<value entered by a user>%')
```

The task allows end users to perform a search of map database objects based on a category and name of the objects. Once the search has been done, the searched objects will be highlighted on the map and their attributes will be available in table format. Results of the real estate objects search can be printed out to a separate web page with or without the map and/or legend.

ASP.NET regular expressions were used to constrain untrusted or uncorrected user input. The regular expression has been created to validate a text field in the panel of the search of an object by its name. All symbols from a to z plus specific Swedish symbols (é,ö,ä,Å,Ä,Å,É) plus “-” symbol and spaces are allowed.

```regex
^[a-zÀ-ÖÇå-ôØ-ëÈ\s]{3,40}$
```

The developed web mapping application is based on the ArcGIS Server platform. Hence, it supports the same set of web browsers and operating systems as any other ArcGIS Server web mapping application does (Table 5).

### Table 5. Web browser & Operating system compatibility of the application (ESRI, 2009a).

<table>
<thead>
<tr>
<th>Web browser</th>
<th>Supported operating systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Firefox 3.0 and</td>
<td>Red Hat Enterprise Linux AS/ES 4.0 Update 2</td>
</tr>
<tr>
<td>Firefox 2.0</td>
<td>Red Hat Enterprise Linux AS/ES 5.0</td>
</tr>
<tr>
<td></td>
<td>SUSE Linux Enterprise Server 10 - Pending</td>
</tr>
<tr>
<td></td>
<td>Sun Solaris 10 (SPARC)</td>
</tr>
<tr>
<td></td>
<td>Windows 2000 SP4 Server, Advanced Server &amp; Datacenter</td>
</tr>
<tr>
<td></td>
<td>Windows 2003 SP2 Server Standard, Enterprise &amp; Datacenter</td>
</tr>
<tr>
<td></td>
<td>Windows Vista SP1 Ultimate, Enterprise, Business</td>
</tr>
<tr>
<td></td>
<td>Windows XP SP2 Professional Edition</td>
</tr>
<tr>
<td>Internet Explorer</td>
<td>Windows 2000 SP4 Server, Advanced Server &amp; Datacenter</td>
</tr>
<tr>
<td>6.0</td>
<td>Windows XP SP2 Professional Edition</td>
</tr>
<tr>
<td></td>
<td>Windows 2003 SP2 Server Standard, Enterprise &amp; Datacenter</td>
</tr>
<tr>
<td></td>
<td>Windows Vista SP1 Ultimate, Enterprise, Business</td>
</tr>
<tr>
<td></td>
<td>Windows XP SP2 Professional Edition</td>
</tr>
<tr>
<td>Internet Explorer</td>
<td>Windows 2000 SP4 Server, Advanced Server &amp; Datacenter</td>
</tr>
<tr>
<td>7.0</td>
<td>Windows 2003 SP2 Server Standard, Enterprise &amp; Datacenter</td>
</tr>
<tr>
<td></td>
<td>Windows 2008 Server Standard, Enterprise &amp; Datacenter</td>
</tr>
<tr>
<td></td>
<td>Windows Vista SP1 Ultimate, Enterprise, Business</td>
</tr>
<tr>
<td></td>
<td>Windows XP SP2 Professional Edition</td>
</tr>
</tbody>
</table>

The application has been tested to work in the Opera 9.x environment as well, because this web browser is also used fairly widely. Some tools worked adequately, whereas other tools were disabled or functioned inappropriately. This test has been done in order to merely check whether the web mapping application can fully function or not. This test has shown that Opera web browser cannot be used to interact with the developed web mapping application properly. However, since the most
popular browsers of 2009 are Microsoft Internet Explorer 6 & 7 and Firefox 2 & 3 (W3schools, 2009), the browser compatibility for the developed application can be treated as appropriate. For internal use within the municipality, where Microsoft Internet Explorer is the de facto standard, this level of compatibility is acceptable and since Firefox is also supported, it will work for the general public as well. While using Microsoft Internet Explorer 7 the best performance of a map redrawing has been demonstrated.

Standard sets of tools cannot fully satisfy the needs of local government and, moreover, the previous web mapping application (cX-Länskartan) in the municipality of Gävle was built based on the ArcIMS platform, includes more advanced functionality, for instance, printing a map image to the Adobe Portable Document Format (.pdf). This way, the customization of the standard set of tools would later be required.

6.5 Evaluation of performance of the web mapping application

The web mapping application with the cached map service includes the municipality of Gävle area. Its rendering in Microsoft Internet Explorer 7 web browser proved to be much faster than the visualization of the same area in an existing ArcIMS-based application. Other implemented tasks provided a better performance as well: search of objects, zooming to search results and map printing – all of them are performed faster than the analogous ArcIMS web mapping application. Several principles have been implemented in order to increase the performance of user-system interaction by using the Microsoft Visual Studio framework:

- The overview map of the web mapping application has been set to refer to another non-cached map service and its static status has been used
- HTTP compression at Microsoft Internet Information Services (IIS) web server has been enabled in order to compress such resources as JavaScript code streamed to the client’s web browser
- Custom DHTML scale has been implemented in the application in order to substitute the standard map scale control and increase the performance of the view redrawing
- Layers in the table of contents are hidden by default in order to increase the speed of application load
- The results of layer identification have been limited to required fields for end users and appropriate aliases have been assigned to their names.

6.6 Customization of the web mapping application

Description of added/customized functionality to the web mapping application is given below (Table 6).

Table 6. Custom tools implemented in the web mapping application.

<table>
<thead>
<tr>
<th>Custom tool</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spatial bookmarks</td>
<td>This tool is the equivalent of the bookmarks tool in the ArcMap application. By using this tool users can quickly move to the predefined geographic area in ArcMap. Spatial bookmarks for the published map document can be added, deleted, or renamed in the ArcMap application.</td>
<td>Adding the tool to a web mapping application can be performed from the ArcGIS Server Manager web application by adding a special task which does not require additional programming; since the tool’s dynamic linked library (DLL) file is distributed freely among ESRI’s user community.</td>
</tr>
<tr>
<td>Tool</td>
<td>Description</td>
<td>Additional Details</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Layout printing tool</td>
<td>This tool is the alternative to the standard tool of printing in the out-of-the-box ArcGIS Server web mapping application. While using a standard tool, users are allowed to create an image of the map with a legend and to render it in another web page. Using this printing tool, users are allowed to export the map image to a .pdf file format and multiple raster file formats and to save it directly on their local disks for printing or further sharing.</td>
<td>While producing some output results (for example, export the map to an image or to a .pdf-file) output files will be stored on a local server hard drive. The procedure of automatic deletion of files after disconnecting a user session should be used. Otherwise, the hard drive will be overwhelmed over time. This option have been implemented in the application as well: once a user session is disconnected, the files are deleted.</td>
</tr>
<tr>
<td>Zoom to scale tool</td>
<td>This tool is the equivalent of the standard tool in the desktop ArcMap application. While using this tool, users are allowed to change the scale of a map by entering a scale value in the box (in the case of vector map service) or choosing a scale level from a drop-down box (in the case of cached map service).</td>
<td>Adding the tool to a web mapping application can be performed in Microsoft Visual Studio or by using Windows File Explorer and any text editor, such as Microsoft Notepad. Adding a tool does not require additional programming since the tool’s .dll file is distributed freely among ESRI’s user community.</td>
</tr>
<tr>
<td>Table results panel</td>
<td>This panel has been designed to substitute the default Results window in the web mapping application. It provides more opportunities to manipulate search or object identification results. Users can sort the columns of the table of results based on a defined field value, or pan/zoom to a found feature in one mouse click. It is possible to select/unselect searched features in order to highlight them, and vice versa.</td>
<td>This panel has been implemented as an alternative to the standard Results panel. However, the current version of the panel has a limitation – it does not support the clearing of the results table and thus, using the standard Results panel for other search tasks could be required. Adding a tool does not require additional programming since the tool’s .dll file is distributed freely among ESRI’s user community.</td>
</tr>
</tbody>
</table>
7 Discussion

Carrying out the project has proved the importance of deep, thorough, and detailed conceptual thinking about the project structure, content, and possible issues. As has been realized, the process of web mapping application development for the public should include a review of several moments.

First and foremost, what GIS data should be delivered to the audience and who is the target audience for the application? Answering this question could help to explain which map layers should be prepared, and in which form they could be accessed by end users so they will be able to best perceive this information. Thereafter, it should be understood which means could best provide wide access to existing map documents or GIS data. This question implies choosing a business logic model and software platform for developing a web application. Proposed logic of the application should comply with organization standards and address needs of end users adequately.

Furthermore, it is required to consider the forms in which the map data will be delivered to end users. Will it be a separate single web mapping application or a map that will be incorporated to a web site of the organization? Understanding these questions could determine the choice of application development platform. Finally, possible obstacles and issues should be considered. Which barriers could emerge during development and implementation of the application? Which hardware and software limitations could impede project realization? Will security issues and data sharing policy produce any problem? All of those questions are worthwhile to think over in advance, even though the real answers presumably will be received only during the project realization stage.

As most conceptual questions have been envisioned within the thesis report, only issues that appeared during this research project will be discussed in this chapter. Throughout the thesis report, numerous issues with developing web GIS applications have been mentioned. What is important to note is that many challenges have appeared to be hardships only during the process of immediate project work. Those challenges are varying and of course are not limited to the ones denoted below.

Implementation of a web mapping application based on an enterprise-wide geodatabase would require appropriate hardware resources. Underestimating the importance of a machine’s capacity could undermine the performance of the whole system. Best practices acknowledge the importance of wise system resources allocation: allocation of hard disk capacity, CPU distribution, and RAM for query processing (ESRI, 2008). The results of this research project have proved the results on the testing performance of retrieving GIS data from DBMS via the ArcSDE interface (Batcheller et al., 2007) and extended some of those testing results to a performance evaluation by using much larger datasets.

Another issue is related to data security matters. Since any interface on a web server content poses a certain risk to the system security, using secured protocols for data transfer and advanced methods of user authentication, and using proxy servers could be required. Even though security considerations were deployed fairly well in the ArcGIS Server platform, while transferring sensitive information – personal profiles of parcel owners or tax information, for instance, – encryption procedures and secured hypertext transfer protocols should be adopted (Zimmermann, 1995).

Some aspects of the use of language-specific characters are worthwhile to notice as well. Working with web GIS software assumes creating multiple files in operating systems, tables in database management systems, and the naming of various system elements. Using specific Swedish symbols for naming any system component could result in unpredictable behavior of the system, particularly in the case of problem with collation mapping in DBMS. Problems with the visualization of text characters (for instance, during the database search) with specific non-Latin symbols (for example, å, ö, ü, é), could appear if end users would use an operating system and a web browser, which does not support required system language packs that are responsible for representing those language-specific symbols.

While carrying out the project, several issues related to ArcGIS Server component interactions emerged. In the municipality of Gävle, the distributed installation of the ArcGIS Server system has been performed (Figure 5). Is this the best schema of ArcGIS Server installation? There is no immediate answer and it will depend upon the municipality’s IT-policy, available hardware resources,
and many others. Thus, the question is whether the ArcGIS Server system should be installed on different machines (distributed installation) or the whole system should be installed on the same machine (single machine installation) in the future. However, this question represents a traditional trade-off. It is harder to administer and monitor a distributed system, but it is easier to allocate resources (that might require high expertise and developing enterprise-wide system design strategies). In contrast, single machine deployment will be easy to administer and troubleshoot, but it will definitely be less flexible from the perspective of resource allocation.

Another issue of web mapping application implementation is that the connection speed of intranet and the Internet is always limited. Thus, this raises the question: how should network resources be allocated? What is the best possible way to allocate network resources according to the number of users and what is the maximum number of users that the network capacity can support? Are there any methods of dynamic network and web server resource allocation according to the number of concurrent users? All of these questions are outside of the scope of the given research, but should be addressed in future work in order to provide appropriate performance of user-map interaction.

Moreover, it was realized that the performance of the developed web application will depend upon the performance of many components. Basically, the performance is defined by the efficiency of both software and hardware parts. Since software use and fine tuning has previously been analyzed, several hardware environment recommendations could be outlined as well. While choosing a hardware configuration, the principle is that the more powerful the hardware is, the better performance can be realized. Below are some recommendations that have been concluded by limited testing of ArcGIS Server application during the research project as well as based on some findings in the literature on the topic.

- It is recommended to use modern double-core central processor units (CPUs) and fast hard disks drives (HDD): Sloan et al. (1992) observed that GIS functions – spatial data editing, for instance – with a heavy load on input/output (I/O) operations are limited by the performance of the disks on which data are stored.
- It might be a good idea to use adequate random access memory (RAM) on the GIS server machine (however, operating systems have limits on RAM, and not more than 2 GB on one (CPU) core is allowed. That is important to understand, because data will be retrieved as fast as a disk can retrieve it, regardless of the software or data model used (Gittings et al., 1991).
- One should consider the option of space allocation: space that is addressed often should be allocated to different physical hard disks. This could partly solve issues of contention for server resources (Batcheller et al., 2007).

During the research, another vital issue in map service implementation has been realized. Performance plays a crucial role in any application development: how satisfied end users are depends on how quickly a map image will be drawn in the end user’s web browser. As has been acknowledged in the thesis report, the level of application performance depends upon the speed of map data retrieval (from the database, for instance). Furthermore, it should be acknowledged that the performance of the developed web application will depend on the speed of data transmission via local or global computer networks as well. Therefore, it is crucial to think about data compression techniques, web caching, and other techniques that can be applied in the case of transferring data over a network, specifically cartographic vector/raster data.

The question of configuring web browsers, web servers, data transmission protocols, and network settings, however, are not within the scope of the given research. For those topics useful instructions and sound theoretical background can be found in the works of Wessels (2001) and Nagaraj (2004). In the case of applying those methods, extensive research on algorithms of web data transmission will be required. Even though computer science and GIS are treated fairly independently of one another, they, actually, are deeply integrated. Indeed, it is supposed that GIS web services could leverage information technology methods. Deployment of any web mapping application can be refined while gaining deeper insights into the problem of GIS data processing from a computer science perspective.
Testing the possibility of the developed application to support concurrent users (up to several hundreds) should be performed in the development environment to make sure that the application will work adequately in the production. Several problems with integrating many data sources (many formats, multiple locations, and various user permissions) could occur and they should not be discounted. Further customization of the application and detailed help system development is highly required.

Another issue is related to the user interface design. In this study, the target audience has been specified in advance: the developed application will be viewed by the public. Hence, it should include only basic tools for map manipulation and map image export. However, should a more advanced web mapping application be developed and what design concept would then be better to apply? Which methods will be the most appropriate for the specific case? Multiple methods of user interface development have been adopted in this research project and thus extended the several authors’ results (Su et al., 2000; Maroney et al., 2007) where tools of the end web mapping application were limited to standard GIS map server out-of-the-box tools.

The process designing the user interface implies that one has to define which methods should be applied in the given case: is it better to create one application that will include all tools required by all target users’ tools and functions or is it better to develop several applications with only specific user group-tailored tools? In order to answer this question, the analysis of how easily users will be able to deal with the application that will have multiple tools, dropdown boxes, menus, etc. should be done (Harris, 2005). Another trade-off is that the maintenance and administering of several applications is a more complex procedure whereas the management of a single application could be a rather simple task. Therefore, user interface should be easy-to-use, but still provide the desired robust functionality.

Concerning the map cache implementation, it is important to take into consideration whether the map cache images should be available to the user immediately or if every action of map interaction (e.g. zooming in/out or panning) should create a new layer cache tile (option “cache on demand”). The first option will let the user access different parts of the map faster, but will have an overall higher workload during cache creation. If the area that should be cached is big, then its caching will take up a huge amount of time. In the second case, the workload during the cache generation will be less, but the access time, in turn, will be bigger since the server will create tiles dynamically. However, all of these questions might be addressed properly only on the production user load, implying the implementation of the application on the production server.

Undoubtedly, caching proved to be a beneficial investment in this project. Moreover, map caching became an industry standard: practically all popular Internet mapping sites use caching. The time that will be spent on deploying the cache will be the time that will be saved by the end users who will consume a web application. However, before one will create a cache for a map service, it is required to answer several questions. What scales do users need to use (to which scales will users be allowed to zoom)? Is the cache for all data in a map document at all scales required or it is possible to cache specific areas of interest (only main towns of the county, for instance)? Questions about the hardware environment are raised as well: is there enough disk space for storing cache image files? Are there enough resources to fill the caching job quickly?

Further issue with cache relates to the cache update. That is, data currency is required for the map content. In this study, where the base map application was developed, a fast map update is not that crucial. Nevertheless, the web mapping application should consistently provide the latest available GIS data. This issue addresses foremost the update of a single map layer of buildings in the whole map cache. As has been envisaged in the report, map cache update represents a multi-stepped process. There are no best solutions and different techniques have to be applied in different situations. In future research projects, attempts to improve the developed process of geographical area-based caching could be performed.

Map service caching has been analyzed extensively within this research project: multiple alternatives have been investigated and tested in the web mapping application whereas in peer reviewed papers developed web mapping application is based on vector data (Simão et al., 2008). In multiple similar
master thesis projects (Harper, 2006; Hongwei, 2002; Hashmi, 2006) no map caching technique overview has been introduced and no attempts to implement map caching was done. In the given research, in contrast, robust methods of map cache generation and update has been implemented. Detailed description of map caching methods and a show of practical results have contributed significantly to the understanding of how ESRI software tools can enhance local government web GIS application studies.

It was realized that the task of implementing the proposed web mapping application to take advantage of the ArcGIS Server platform can be technically challenging. ArcGIS Server framework integrates several software systems, which have been installed on several machines in the municipality of Gävle. Furthermore, to migrate a web mapping application from a development server to a production server in the distributed installation model is difficult. Therefore, since the use of existing programming code samples to implement several custom tools were required, the transferring of those tools to a production server will produce more problems with their integration into the final application.
8 Conclusion

During the thesis work a web mapping application of the base map cX-Länskartan of the municipality of Gävle was developed. The created application allows users to access map data sets and query needed information quickly and efficiently. The web mapping application proved that ESRI ArcGIS Server had the capacity to develop a web mapping application that could allow the general public to obtain the required map information.

The web mapping application was implemented on top of ESRI’s ArcGIS Server (version 9.3) and ArcSDE (version 9.3) for geodata storage and ArcGIS Desktop (version 9.3) for map document compilation. Map data are stored in the Microsoft SQL Server (version 2005) database. ArcSDE in this case is a gateway to the DBMS for the web GIS application. For developing custom tools and for refining the user interface, Microsoft Visual Studio ASP.NET was used (version 2008) which enables the development of map-based applications using state-of-the-art object-oriented environments, such as C# or .NET. A number of freely distributable Microsoft .NET programming samples developed by the ESRI user community were embedded in the final web mapping application.

Microsoft Visual Studio proved to be a robust platform for developing and customizing web mapping applications because it provided a framework that could integrate models and program classes which may have been written by using various programming languages. Moreover, the platform provides the possibility of immediate deployment of the built Microsoft .NET objects in web mapping application. The current platform configurations were as follows: web server: Microsoft IIS 5.0; OS: Windows XP Professional SP2; RAM: 1 GB; HDD: Fujitsu 100 GB 8200 RPM; DBMS: Microsoft SQL Server 2005; GIS software: ArcGIS Desktop 9.3, ArcSDE 9.3, and ArcGIS Server 9.3.

The aims of the given project were fulfilled: ESRI ArcGIS Server framework was used in order to employ several particular strategies for web mapping application development, design, customization, and performance optimization. As has been envisioned in the report, web GIS application design, development, and implementation are inevitably related to multiple challenges. From the perspective of web GIS application, a developed base map web application is very beneficial because many users will find it easier to get the information they need via a web application rather than by a visit to the municipality. Moreover, other government agencies, municipalities, and private companies can use this resource. It was confirmed by scientific literature and business papers that web applications in local government provide wide and open access to map information to citizens and save its users time and money.

Different methods of map document have been identified as well: creating attribute and spatial indices for map layers, map symbology simplification, and scale dependency in layers redrawing should be employed. Concerning map caching techniques, the generation of map cache image tiles for the areas that are expected to be visited more often than other areas is required. Other area cache will be created dynamically as users will explore those geographical areas by using cache on-demand techniques. Several cache generation and automatic cache update models have been developed as well.

At the very moment, the developed web mapping application is a proof-of-concept implementation. Various usability tests and a test of the effectiveness in a quasi-production environment are required in order to improve the proposed application content, performance, and functionality. Presumably, applied techniques for the development of the web mapping application prototype will be refined and will be adopted for the development of the production web mapping application for the whole county of Gävleborg.
9 Future prospects

Plans for the future could include developing additional mapping modules and extensions for a developed web mapping application with rich GIS functionality. The current web mapping application functionality can be extended by linking other existing GIS applications already in place at the municipality of Gävle. Different ESRI software and services, including Arcweb™ Services should be evaluated in order to provide more superior map functionality and rich map content. Finally, powerful tools for data mining from RDBMS could be developed and the integration of existing databases and web mapping application interface could be done as well.

Current technology for web application development is expanding to include the use of dynamic web applications with many features for users that is referred to as Web 2.0. Composite web applications are also called Mashups, and the developed web mapping application could integrate maps from different resources, for example, Google Maps, Microsoft Virtual Earth, or Yahoo! Maps as well as include data from other agencies’ map services. For these purposes, different programming interfaces could be used: ArcGIS JavaScript API, ArcGIS Flex API, or the recently announced Microsoft Silverlight™/WPF™. All of these interfaces allow developers to implement rich internet applications with the capabilities to integrate data from different sources and of different content – both spatial and non-spatial (Minarčík & Őzana, 2009).

In such applications, huge amounts of data sources and functionalities of existing as well as future web applications and services may be easily combined. Furthermore, better integration with Google Earth and Microsoft Virtual Earth applications would be required since in ArcGIS 9.3.1 users of ArcGIS Desktop will have integrated access to satellite imageries from Microsoft Virtual Earth. Developers of municipal web mapping applications could benefit from this technology and it could make it possible to create web applications with advanced GIS capabilities and minimum programming and time requirements.

Any maintenance of a web mapping application would require monitoring the web server and client interaction as well as collecting statistics on the use of web mapping application, server resource allocation, and traffic load. One of the most widely acknowledged tools for web traffic monitoring is the open-source application Jakarta JMeter (accessible for download on the web site http://jakarta.apache.org/site/downloads/downloads_jmeter.cgi). This tool has been widely accepted in the computer science community for testing the performance of web applications. As a future project, applications of this tool in web mapping application performance evaluation could be found.

Another useful tool that has been widely used by the ESRI GIS user community for monitoring the performance of geodatabase data fetching is Geodatabase Toolset (accessible for download at http://www.esri.com/software/arcgis/extensions/gdbt/index.html). This tool includes system analysis tools and functions that might help to diagnose certain performance issues. It could be used in order to improve maintenance of ArcSDE geodatabases. Applying it to a web mapping application, it provides the ability to generate reports by using diagnostic tools, which could supply database administrators with DBMS statistics and spatial indexing information. The time of data retrieving from the non-cached vector-based dataset could pay a crucial role in the user-map interaction. Therefore, information such as relative data fetch time and data refresh time (per layer) could be supplied based on the analysis results. Unfortunately, the last version of this application supports only the ArcGIS Desktop 9.2 environment and cannot be used in the research project. However, the ESRI Product Team is currently working on a 9.3-compatible release and presumably a new version will be available soon, meaning that the toolset could be applied in future research.

ArcGIS Server 9.3 introduced ArcGIS JavaScript API for quick and easy development of web mapping applications. The possibilities of ArcGIS JavaScript API coupled with ArcGIS Server REST API provide a simple and open web interface to GIS services hosted by ArcGIS Server. The capabilities of these technologies should be elaborated in detail in future work and, if applicable, be applied in the development of web mapping applications for municipal use. To see an example of this,
a web page code created for the municipality of Gävle ArcGIS Server map service by using ArcGIS JavaScript API is in the Appendix A.

New ways of delivering map information to users should be analyzed in future research as well. OGC data service standards of delivering information that can be combined with multiple data sources should be taken into account. Developing OGC services in order to support an interoperable environment with advanced GIS functions of map data manipulation could be considered. The use of OGC web map services is becoming an industrial standard and more Swedish municipalities use these services for data sharing via the Internet. Thus, their use is common enough to adopt this technology as an additional required format for future web mapping standards within the municipality of Gävle.

Further optimization of the system, if applicable, should be according to the considerations of network features, server loads, as well as the administrative needs involved in configuring and maintaining the developed system. Moreover, testing has been performed in the development environment with a limited user load. An obvious extension to the current work is the evaluation of the application performance in a production environment with increased user loads, again, implying the implementation of the application on the production server.

To conclude, the ArcGIS Server platform tends to be more open and the wide support of Java development framework supports this statement. The Java Eclipse framework can be used with Visual Studio for developing web mapping applications. The use of open standards in map application development and support for different operating system platforms can be seen as well. In the future, even more opportunities for developers will be created. A greater number of user friendly interfaces for developers will be released by major IT and GIS vendors. Finally, complex GIS functionality will be possible to implement in web applications even without having GIS or programming language expertise, which would allow to developers to take advantage of web benefits for the GIS community.
References


Tikunov V. (2004), *An Introduction to Geoinformatics*, Moscow State University Press, 729 p. [In Russian language]


Appendix A

Web page with web mapping application interface:

The code for the web page with an integrated ArcGIS Server REST API map service is below. The code consumes elements of ArcGIS JavaScript API via the Internet.

```html
<html>
  <head>
    <title>Gävle Kommun: cX-Länskartan</title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
    <script type="text/javascript" src="http://serverapi.arcgisonline.com/jsapi/arcgis/?v=1.3"></script>
  </head>
  <body>
    <div id="mapDiv" class="tundra" style="width:900px; height:600px; border:1px solid #000;"></div>
    <h4>Cx-Kartan</h4>
  </body>
</html>
```
Appendix B

This script has been used to automatic update all cache tiles in an ArcGIS Server 9.3 map service cache by using Windows Schedule Task tools. The script is run in ArcGIS Server 9.3 environment and is based on Python 2.4.

```python
# Here we access the geoprocessing tools
import arcgisscripting
gp = arcgisscripting.create()

# Here we set up all of the variables which are used in the
# geoprocessing update tool
server_name = "studio2008"
object_name = "cx_karta"
data_frame = "Layers"
layers = ""
constraining_extent = ""
scales = "64.000"
update_mode = "Recreate All Tiles"
thread_count = "1"
antialiasing = "ANTIALISING"
update_feature_class = "byggnade.sde"

# Here the Update Map Server Cache tool is being run
try:
    print 'Starting Cache Update'
    gp.UpdateMapServerCache(server_name, object_name, data_frame, layers,
                             constraining_extent, scales, update_mode, thread_count, antialiasing)
    print 'Finished Cache Update'

# If the tool will fail, then the error messages could appear
except:
    gp.AddMessage(gp.GetMessages(2))
    print gp.GetMessages(2)
```

This geoprocessing model was developed during the research project to update cache tiles automatically based on the extent of the edited objects in the buildings layer of the versioned ArcSDE geodatabase. To run the model, the custom geoprocessing tool is required (available for download from http://arcscripts.esri.com/details.asp?dbid=15703).
Here is the Python script of the above shown geoprocessing model:

```python
# Import system modules
import sys, string, os, arcgisscripting
# Create the Geoprocessor object
gp = arcgisscripting.create()
# Check out any necessary licenses
gp.CheckOutExtension("3D")
# Load required toolboxes...
gp.AddToolbox("C:/Program Files/ArcGIS/ArcToolbox/Toolboxes/Geodatabase Tools.tbx")
gp.AddToolbox("C:/Program Files/ArcGIS/ArcToolbox/Toolboxes/Server Tools.tbx")
gp.AddToolbox("C:/Program Files/ArcGIS/ArcToolbox/Toolboxes/Analysis Tools.tbx")
# Local variables...
Version_difference = "C:\GIS_data\Reconcile_tool\output1.shp"
CX_KARTA_DBO_PK_BYGGNADSYTA = "Database Servers\TYCOON-A9D21733_SQLEXPRESS.gds\CX_KARTA (VERSION:dbo.DEFAULT)\CX_KARTA.DBO.PK_BYGGNADSYTA"
Buffered_contour = "C:\GIS_data\Reconcile_tool\output1_Buffer.shp"
Output_Host = ""
Output_Map_Server = ""
# Process: Show Edits Since Reconcile...
# Process: Buffer...
gp.Buffer_analysis(Version_difference, Buffered_contour, "50 Meters", "FULL", "ROUND", "ALL", "")
# Process: Manage Map Server Cache Tiles...
gp.ManageMapServerCacheTiles_server("TYCOON-A9D21733", "Byggnade_updating", "Layers", "CX_KARTA.DBO.PK_BYGGNADSYTA", "125000;64000;32000;16000;8000;4000;2000;1000", "Recreate All Tiles", Buffered_contour, "2", "NONE", Buffered_contour, "IGNORE_COMPLETION_STATUS_FIELD")
```