GIS PLATFORMS FOR WINCE

Student: Markus Pettersson

Master thesis 20p, D-level
Department of computer science and electronics
Mälardalen University

Supervisor: Jonas Neander
Examiner: Mats Björkman

ABSTRACT

This master thesis investigates, on mission from CC-systems, the possibilities to use Windows CE 5.0 as a platform for Geographic Information System (GIS) with CC-Systems onboard computer CCP XS. Existing software based on PC technology cannot easy be translated and used in a system with Windows CE.NET. The purpose of this master thesis is, to find and further adapt a suitable software or component for GIS. The final goal is to write a demonstrator/prototype.

The survey of development options and existing components for GIS and Global Positioning Systems (GPS) for Windows CE show that there exist several candidates for the CCP XS platform. We will show that the most suitable architecture is .NET Compact Framework (.NET CF) and the most suitable component is Fransons GPS tools. .NET CF makes it possible to write managed code and the tools available for .NET CF render a more powerful programming possible. Franson GPS tools fulfil the requirements and have, compared with its competitor additional benefits.

We have in this thesis limited ourselves to find a component that fulfils some desirable features and write a simple demonstrator in C#, demonstrating these features. The component should have support for ESRI shapefiles[1] and the Swedish reference system RT90. It should be possible to view a map in a suitable raster format and receive a position from the GPS.

SAMMANFATTNING

Det här examensarbetet undersöker, på uppdrag av CC-Systems, möjligheterna med att använda Windows CE 5.0 som en plattform för Geografiskt Informations System (GIS) på CC-Systems fordonsdator CCP XS. Befintlig mjukvara baserad på PC teknologi kan inte enkelt överföras och användas i system med Windows CE.NET. Avsikten med det här examensarbetet är att hitta och anpassa en mjukvara eller komponent för GIS. Det slutgiltiga målet är att implementera en demonstator/prototyp.

Undersökningen av utvecklingsalternativ och existerande komponenter för GIS och Global Positioning System (GPS) för Windows CE visar att det finns flera kandidater för CCP XS plattformen. Vi kommer att visa att den mest lämpliga arkitekturen är .NET Compact Framework (.NET CF) och den mest lämpliga komponenten är Fransons GPS tools. .NET CF gör det möjligt att skriva managed kod och verktygen som finns för .NET CF möjliggör en kraftfullare programmering. Franson GPS tools uppfyller kraven och har i jämförelse med sina konkurrenter även andra fördelar.

---

[1] Environmental System Research Institute (ESRI) shapefile: a file format for vector graphic.

Vi har i det här examensarbetet begränsat oss till att hitta en komponent som uppfyller några utvalda och önskvärda egenskaper och sedan implementerat dessa i en demonstator i C#. Komponenten ska ha stöd för ESRI shapefiler[2] och det svenska referenssystemet RT90. Det ska vara möjligt att visa en karta i lämpligt raster format och ta emot en position från GPS:en.

---

[2] Environmental System Research Institute (ESRI) shapefile: ett filformat för vektor grafik.

	Student Markus Pettersson		Doc Nr		Säk klass
	Supervisors on CC-System Johan Persson and Jörgen Martinsson		Date 2006-05-15	Rev	Filename GIS platforms for WinCE

CC SYSTEMS

TABLE OF CONTENT

PREFACE

Many thanks to CC Systems were I accomplished my master thesis. I was introduced to a subject that I knew very little about. Admittedly did my task sound very interesting, like a big challenge and fun right from the beginning, however the more I learned about the current subject the more fun and interesting it became. Thanks to my master thesis I found a new area that I gladly and hopefully will enter more deeply in my future career.

Thanks go to my supervisors Jörgen Martinsson and Johan Persson at CC Systems and Jonas Neander at Mälardalen University.

Special thanks go to my friends, family and my girlfriend that supported me during the project. Your encourage words has been very valuable for me.

Student		Doc Nr		Säk klass
Markus Pettersson				
Supervisors on CC-System		Date	Rev	Filename
				GIS platforms for WinCE

7 (76)

1 INTRODUCTION

CC systems is a company that offers control and information system solutions for machines and vehicles in rough environments, for example in military vehicle and forestry machines. Their head office is sited in Alfta, Sweden, were activities as development, production and deliveries are performed. The company was founded in 1991 and the activity was then limited to Alfta. Gradually the company has expanded their (development) activity to Örnsköldsvik, Uppsala, and Västerås. CC Systems products include different displays and on-board computers, I/O- modules and controllers, application modules and communication modules. The premier on-board computer is a large pc based version named CCP XL and recently a new model CCP XS has been developed. The new on-board computer use the same technology as a modern PDA and the reaction from customer, and hypothetical customer on this product has been positive. One advantage of this new product is the lower price relative to a pc-based computer. For users/costumers, which purpose is to use the on-board computer for simple functions, such as positioning, showing maps and video imaging, this computer is to prefer. Customers have also made inquiries about functions concerning Geographic Information System (GIS) that they like to have in this computer. CC Systems is therefore interesting in investigating the possibilities to be able to use CCP XS as a platform for GIS [1, 2].

1.1 Problem formulation

The on-board computer CCP XS use Windows CE.NET (5.0), an operating system for mobile devices or PDAs. Existing software based on PC technology cannot easily be transferred and used in systems with Windows CE as platform. This thesis will focus on investigating the possibilities of using the on-board computer as a platform for GIS, i.e. find suitable software and bring it into line with consumer needs.

1.2 Goal and purpose

The purpose of the master thesis is to find and further adapt a suitable software or components to illustrate the possibility of using the on-board computer CCP XS as a platform for GIS. The goal is to develop a demonstrator, which can attract customers, and show that CCP XS can be extended for further use.

1.3 Delimitations

The minimum requirements for this thesis were stated in the beginning of the project. We also stated desirable functionality, implemented if, time allowed.

Minimum requirements:

It should be possible to view a map in a suitable format for raster pictures, i.e. GIF (Graphics Interchange Format). Support for ESRI shapefiles, which is a common vector format in GIS.

The demonstrator should be able to receive the current position from the build in GPS and present it on a map and have support for the Swedish reference system RT90.

Desirable or bonus functionality:

Show isohypses (Swe. höjdkurvor), simular to handle and show layer, show waypoints, track recording, transmit and receive information and the ability to use "Lantmäteriets friluftskartor (.RIK)"

1.4 Outline

The report should preferable be read in chronology order, Chapter 2 and 3 contains fundamental information about Geographic Information Systems (GIS) and the on-board computer CCP XS respectively. Chapter 4 presents Windows CE and the connection to closely related systems.

Chapter 5 present and summarize possible development alternatives, with pros and cons, for Windows CE.

Chapter 6 begins with an introduction to Global Navigation Satellite Systems (GNSS) where a detailed description of the Global Positioning System (GPS) follows. GLONASS, a Russian GNSS and a European system under development called Galileo are also presented. The chapter ends with a description of the NMEA 0183 protocol for transferring of positioning information.

The next chapter presents geodesy. Geodesy is the science concerning the size and shape of earth and how to determine positions on it. We describe the fundamental terms for understanding the connection between geographic coordinates (the coordinates received from a GPS) and the planar coordinates used in flat maps.

Chapter 8 describes the raster and vector data format, used to represent data in a GIS

A survey over the most suitable components and software so far are presented in chapter 9. The chapter ends up with a comparison between the products.

In chapter 10 a discussion around the survey is conducted and chapter 11 presents the demonstrator, which choice was made concerning component and how the tasks and requirements have been solved.

In chapter 12 we end the thesis with a discussion around the demonstrator.

Student		Doc Nr		Säk klass
Markus Pettersson				
Supervisors on CC-System		Date	Rev	Filename
Johan Persson and Jörgen Martinsson		2006-05-15		GIS platforms for WinCE

CC SYSTEMS

2 INTRODUCTION TO GIS

2.1 What is Geographic Information System (GIS)?

A Geographic information System is a system that somehow handle spatial referenced data, it can be further explained as a technology used to collect, analyse, manipulate and view data based on geographic information. A word frequently associated with GIS is a map, which most often is, but not necessary have to be a component of GIS. Since GIS use spatially referenced data maps are efficient tools to view information with and therefore are maps often associated with GIS.

The data or information in a GIS is linked to specific positions on the surface of earth and different layers holds data about a particular kind of feature. For example, all roads can be represented in one layer, all lakes in another, i.e. any feature can be mapped into a specific layer to represent a certain area and theme. Several layers can be combined for analysis and formation of new layers etc. To further explain GIS I would like to quote Jeffery Star and John Estes definition of GIS[3]

"A geographic information system is an information system that are designed to work with data referenced by spatial or geographic coordinates. In other words, a GIS is both a database system with specific capabilities for spatially-reference data, as well a set of operations for working with data…In a sense, a GIS may be thought of as a higher-order map."

2.2 Mobile GIS

Mobile or field GIS is an extension of a Geographic Information System (GIS). Mobile GIS enables people in the field to capture, update, manipulate, analyse and display geographic information.

Mobile GIS integrate at least one of the following technologies:

- Mobile devices.

- Positioning determining equipment such as GPS.

- Telecommunication service provider, wireless connection Internet access.

- GIS content provider.

---

[3] Geographic Information Systems:
An Introduction (Englewood Cliffs, NJ: Prentice-Hall, 1990), page 2-3

Figure 1 depicts the structure of mobile GIS and explains how the different components interact [3, 4, 5].
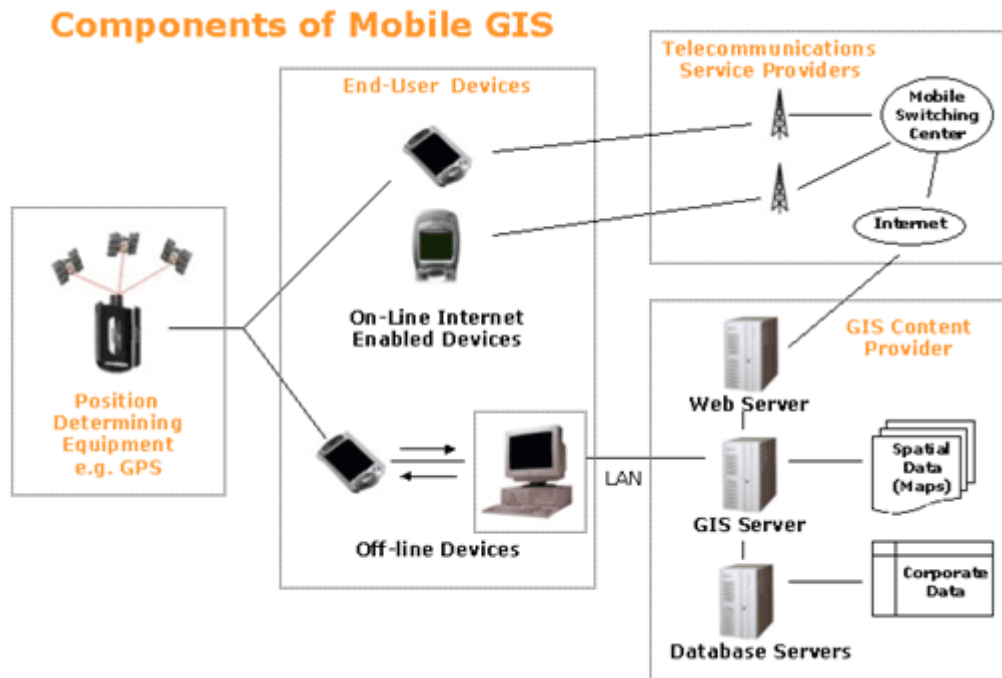
Figure 1. Components of mobile GIS [[51]

3 THE ON BOARD COMPUTER CROSS COUNTRY PILOT XS (CCP XS)

3.1 Overview

The CCP XS (Figure 2), developed by CC Systems, is a compact on board computer for Windows CE.NET and Linux. There are built in modems for GPRS/GSM[4], WLAN[5] and bluetooth[6] communications. The on board computer also offers a built in GPS receiver, these features together makes it ideal as a mobile client for transportation and logistics. Further is CCP XS used in control systems for machines and vehicles, where the on board computer can work as a control computer and Man Machine Interface (MMI). MMI means that the user can control and receive information from the system through input and output on the display

Figure 2. The on board computer CCP XS

CCP XS support CANopen, a protocol for communication buses that are used in many machines.

The CPU used in CCP XS is an ARM processor of type Intel Xscale, a high performance processor with low power consumption. A non-rotating hard disc (compact flash) is used to make it durable and well suited for tough environments [2].

---

[4] GPRS/GSM (General Packet Radio Services /Global System for Mobile Communications): Where GPRS use GSM network to transmit data.

[5] WLAN (Wireless Local Area Network)

[6] Bluetooth: A wireless technology for short range transmission of data

4 WINDOWS CE

4.1 Overview

Windows CE is a 32-bit, operating system designated for embedded systems. The work with Windows CE began in 1993 and the first version was released in 1996. In 1998, the second edition of Windows CE was available for the market, this time in a scalable version. The developers could now choose the components to include. Some of the most important criteria's for Windows CE were, and still are that the operating system should run on many different processors and it should be component based. To make it easier for desktop developers to make application in Windows CE, the initially demand were that WIN32 API should be available. Windows CE should also be a real-time operating system.

The latest version of the operating system is Windows CE 5.0. Updates concerning tools, new technologies and support for more processors are some of the things that have changed from introduction and earlier versions. In Windows CE 5.0 it is possible for developers to write applications for WIN32, MFC and the .NET Compact Framework. Windows CE processor support is extended to ARM, SHx, MIPS and x86. All since Windows CE version 4.2 some of the source code been available as shared source. The latest version of Windows CE extends the shared source code even further.

Platform builder is an Integrated Development Environment (IDE). It is used to develop a custom platform based on the core of Windows CE. Platform builder contain tools that is needed to be able to configure, build, load and debug the operating system to fit the needs and hardware.

4.2 Windows CE, Pocket PC and Windows Mobile

Sometimes it can be hard understand the differences between Windows CE, Pocket PC, Smartphone and Windows mobile. Windows CE is the core OS of all the others and Windows Mobile is a new brand for a subset of Windows CE based platforms, including Pocket PC, Pocket PC Phone edition and Smartphone. Table 1 shows the relation between Windows CE and custom platforms.

Handheld PC 3.0	Built on Windows CE 2.1
Handheld PC 3.1	Built on Windows CE 2.12
Pocket PC 2000 Handheld PC 2000 Pocket PC 2002 SmartPhone 2002	Built on Windows CE 3.0
Pocket PC 2003/Windows Mobile 2003 Pocket PC 2003 Second Edition (SE) / Windows Mobile 2003 SE	Built on Windows CE 4.2
Windows Mobile 5.0	Built on Windows CE 5.0

Table 1. relation between Windows CE custom plattforms

The difference between the custom platforms based on the core of Windows CE is the composition of components. The platforms under Windows Mobile consist of a defined set of minimum requirements that should be fulfilled [6, 7, 8, 9, 10].

5 Development alternatives

5.1 Managed Code vs. Native

To be able to choose development tools and the type of code, managed or native, to write for the application, it is necessary to examine the different alternatives, pros and cons. Managed code is not specific to one software platform and processor but written to a set of run-time environment APIs while native is launched to a specific one.

Managed code

The source code written in .NET is compiled into a language called Common Intermediate Language (CIL) by a compiler that supports .NET. CIL (also called Intermediate Language (IL) and Microsoft Intermediate Language (MSIL)) can be seen as pseudo code or a middle state for the code before it is compiled into native code on the target computer. Information about the code, such as data types is saved as metadata. CIL and metadata is together called managed code and is saved in an assembly. When the assembly should execute on the target computer it is compiled into native code by a Just In Time (JIT) compiler, that translate the CIL instructions in the assembly to platform specific instruction (Figure 3).
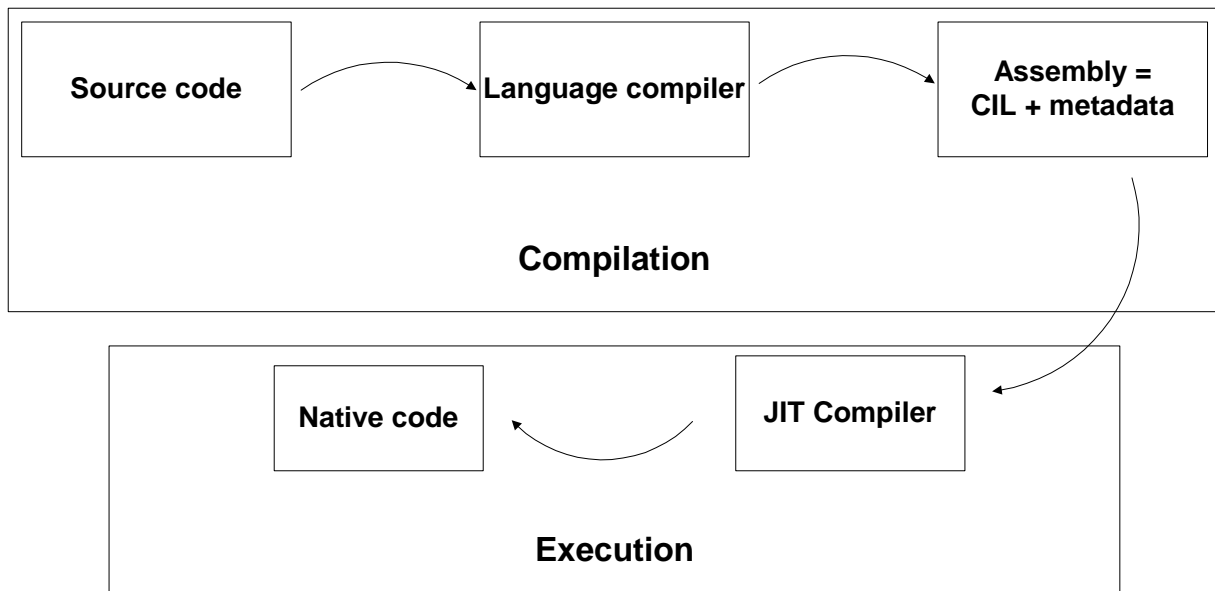
Figure 3. Compilation of .Net based source code

There are many advantages in writing managed code applications. There are powerful development tools, such as C#, which lead to fast developments. Since the code is not specific to a particular hardware or software platform it reach more people, and it is easier to maintain the code and by those means easier and cheaper to support the customers with new or improved functionality. It is enough to change or improve one set of code instead of many which would be necessary in case of programming the application in native code. Another

great benefit of managed code is that it creates a safe and robust code. A big problem for developers is and has been memory management. A majority of bugs is deduced from mistakes with memory management and memory leaks. In .NET developer does not need to think of memory leaks since the garbage collection automatically collect and return the used memory, this is included and handled by the .NET run-time environment called Common Language Runtime (CLR), which is one of the two main parts of the .NET Framework.

.NET Framework consists of:

- Class Library
- Common Language Runtime (CLR)

The class library is common for all programming languages in .NET. Developers are no longer forced to choose a programming language depending on its class library or API. They can choose language more freely without limiting the application. The class library is also used for different kinds of applications, i.e. windows forms, web forms etc. It is also possible to mix different .NET languages in a project.

The CLR is based on the Common Language Infrastructure (CLI) specification, which is an international standard for creating development and execution environments in which languages and libraries work together seamlessly. CLR load and verify the code, during time for execution is services concerning security, allocation of memory and garbage collection offered.

Some of the cons of managed code are for example during that all threads are suspended during garbage collection. This could in worst-case lead to interruption up to several seconds were the user could not use the application. There are also some limitations in the .NET Framework, and especially in the .NET Compact Framework (.NET CF), which is a subset of .NET Framework except from some unique classes that are not included in .NET Framework. .NET CF is used in systems with limited resources, for example in mobile devices and PDAs (.NET CF will be explained in more detail further on). The limitations, both in .NET Framework and .NET CF sometimes require calls to native code. Managed code also execute slower than native code, which should be considered in development of applications with high demands on fast execution.

Native code

Native code is written to a specific set of software platform APIs and specific CPUs. It offers higher performance than managed code. Since applications developed with native code only can run on specific devices and software platforms this limit the use and complicate the support work, e.g. if an application should be distributed to many different platforms, it is not enough to maintain one code set. Native code also requires the developer to write its own security and management code, which are handled automatically by CLR in managed code. Tools for embedded systems have always been one step behind the tools for desktops. This has changed

with .NET and therefore is not the old tools for native code development as powerful as the one for managed.

5.2 Available APIs and development tools

The alternatives of Windows CE .NET programming are represented by four different Application Programming Interfaces (APIs), Microsoft Win32 API, the Microsoft Foundation Class (MFC) library, Active Template Library (ATL) and the .NET Compact Framework (.NET CF).

The Win32 API and MFC are used for native code development and use the programming language C/C++. Applications can be developed using embedded Visual C++ 4.0 or Visual Studio 2005. ATL is a set of template-based classes, which primary is used to build Component object Model (COM) objects and not applications. COM is therefore not suitable in this thesis. .NET Compact Framework is used for managed code development. Application targeting the .NET CF can be developed using Visual Studio 2003 Professional or Visual Studio 2005. These two versions of Visual Studio support smart device development using VB.NET and C#.NET. The two languages, VB.NET and C#.NET, cannot be combined in the same project file, this should be considered since it is possible when developing desktop applications.

5.2.1 Strengths of the APIs

This table present the strengths for Win32, MFC and .NET CF.

Win32	MFC	.NET CF
Smallest and fastest exe files and DLLs	Object-oriented. Inheritance, Encapsulation, Polymorphism.	Object-oriented. Inheritance, Encapsulation, Polymorphism.
Lowest memory overhead	Container classes supporting arrays, lists, object maps and simplicity data handling.	Container classes supporting arrays, lists, hash tables, dictionaries, and stacks.
Required for device drivers	Good tool support. A set of wizards assists in adding message handlers to windows, virtual functions, adding forms, and adding classes.	Powerful tool support— Integrated Forms Designer makes it easy to drag & drop items from a toolbox; code behind UI elements is automatically generated.
Required for control panel applets		Namespaces.
Required for shell extension		Automatic garbage collection eliminates memory leaks.
Does not		Portable machine instruction

		set, MSIL / CIL, provides binary portable of executable (.exe & .dll) files.

5.2.2 Weaknesses of the APIs

This table present the weaknesses for Win32, MFC and .NET CF

Win32	MFC	.NET CF
Object cleanup is the responsibility of the application/driver programmer, making this API prone to memory leaks.	Object cleanup is only semi-automatic, therefore less prone to memory leaks than Win32, but still somewhat vulnerable since MFC is thin wrapper on top of Win32.	Overhead of calls between managed and unmanaged code is high.
Low-level API	Binary size	Binary size.
Procedure-oriented API, not object-oriented.		COM Interoperability: Requires writing Win32 wrappers that call the COM interface functions.

In common for all the APIs mentioned in sub section 5.2 is that they are subsets of their equivalences in the desktop versions for Windows. But only .Net Compact framework has received a unique name. The others, MFC and Win32 have the same name in Windows CE as in the desktop versions.

5.3 Windows 32-bit (Win32)

Win32 is a low level API on which all the other APIs rely on in the end.

5.4 Microsoft Foundation Classes (MFC)

As a consequence of the difficulties targeting a low-level API involve, MFC were created as an object-orientated layer on top of Win32.

5.5 .NET Compact Framework (.NET CF)

The .NET Compact Framework is said to be a subset of the .NET Framework, which is available for desktop computers. It would however be more correct to say that .NET Compact Framework contains a subset of classes from the .NET framework, since it also has some exclusively designed classes that are not included in .NET Framework. There are classes that concern infrared technology, specific SQL Server CE classes and Microsoft.Windows.CE classes and those are only available for the .NET Compact Framework.

One prerequisite to fulfil the words often associated with .NET "write once run anywhere" is that the .NET Framework or in this case, the .NET CF is installed on the computer. The .NET Compact Framework inherits the full .NET framework architecture of the common language runtime and managed code execution.

The .NET compact framework

- run programs that are independent of hardware and operating system

- supports common network protocols, and connects seamlessly with XML Web services.
- provide developers with a model for targeting their applications and components to either a wide range or specific category of devices.

- provide benefits of design and optimisation of limited system resources.

- obtain optimal performance in generating native code using just-in-time (JIT) compilation [10, 11, 12, 13, 14, 50].

6 GNSS – GLOBAL NAVIGATION SATELLITE SYSTEM

6.1 Overview

GNSS is the general term for a satellite based navigation system, which includes one or more constellations of satellites. There are at the moment two systems in use (GPS and GLONASS) and one is under construction (Galileo).

6. 2 GPS

6.2.1 Overview

GPS or actually NavStar GPS (Navigation Satellite Time and Ranging Global Positioning System) is a worldwide radio-navigation system where you can receive your current position through a GPS receiver. The system was originally developed by the U.S. department of defence (DARPA) and refers to ground stations, receivers and a set of satellites constantly circling the earth. The aim with the GPS was initially an application for military purpose. Luckily in the 80s, the use of GPS expanded and became available for civilian use as well. The work with the GPS system proceeded and was fully extended for the civilian market in 1993. There are three segments of GPS (Figure 4).
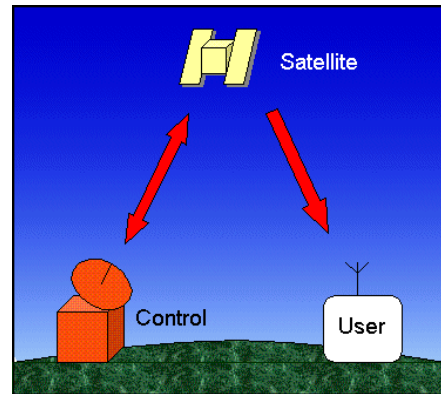

Figure 4. The segments of GPS

6.2.2 The space segment

The space segment includes the satellites, it should be at least 24 of them (29 satellites in Feb. 2005), which orbit the earth in six orbital planes every 12 hours, or more precise every 11 h and 57,97 minutes. They operate at high altitude to be able to transmit radio signals to a large area of the surface of earth. The number and constellation of the satellites mean that at least 4 satellites are available all around the world 99.9 % of the time. But since the radio signals are low powered it is important to have free sight to the receiver. The signals are strong enough to pass through glass, plastic and clouds, but not buildings, rocks etc. The inclination is 55 degrees. This is about equal to the south of Bornholm.

6.2.3 The control segment

The control segment consists of five control stations sited at various places around the world. These stations calculate the positions for the different satellites and a master control station update the satellites with orbital and clock corrections.

6.2.4 The user segment

The user segment includes the user and the GPS receiver. To be able to calculate the three-dimensional position at least four satellites must be in direct sight and for a two-dimensional position three satellites must be in sight. The position is calculated trough triangulation. Due to the inclination it is easiest to receive a position if there is free sight to the south.

6.2.5 Principle for measurements

The principles for GPS measurement can be divided into two main types, absolute and relative measurements.

6.2.5.1 Absolute measurement

The position is determined with help from satellites and one receiver (Figure 5). Calculations concerning the distance to the satellites are computed in the receiver. The message from the satellites contains the satellites orbital parameters, which makes it possible to determine their position. The distance together with the position of the satellites makes it possible to determine the current position.
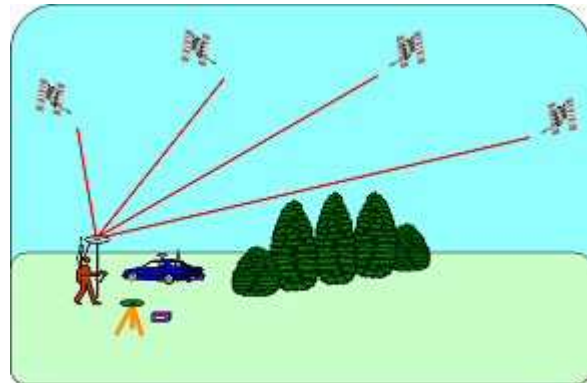

Figure 5. Absolute measurement [21]

6.2.5.2 Relative measurement

Relative measurements are used to attain a higher level of accuracy than obtained from absolute measurement. The principle for relative measurements is that the position is determined relative a "known" spot. Two receivers are a minimum to be able to perform relative measurements. One receiver work as a reference station with a known position and the other as a mobile unit (Figure 6). The reference station can compare the measured distance with the real distance to the satellite and send this to the other receiver as correction or differential data. This makes it possible for the mobile unit to adjust the calculation and receive a more accurate position. 4-5 satellites are needed in order to perform relative measurement calculations.
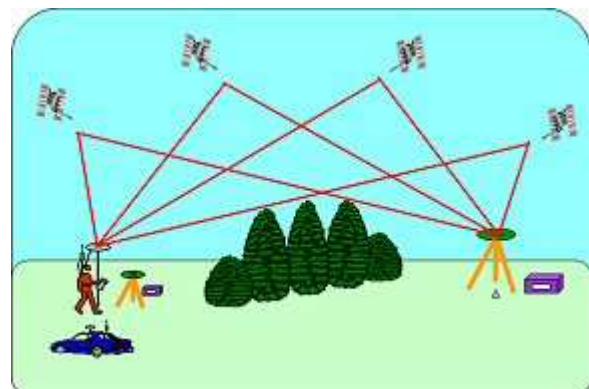

Figure 6. Relative measurement [21].

| Student
Markus Pettersson | | Doc Nr | | Säk klass | |
| Supervisors on CC-System
Johan Persson and Jörgen Martinsson | | Date
2006-05-15 | Rev | Filename
GIS platforms for WinCE |

CC SYSTEMS

6.2.5.3 Determination of the position

We mentioned a method called triangulation, to determine the receiver position in section 6.2.4. In this section we will describe the method in more detail [16].

Figure 7: When the position and distance of <u>one</u> satellite is known the position of the receiver can be determined to be somewhere on the imaginary surface of the sphere with:
- Radius = distance
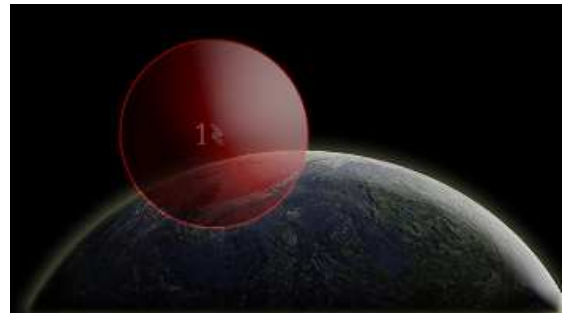- Origin = satellites position

Figure 7. One satellite [16]

Figure 8: When the distance of <u>two</u> satellites is determined and their positions are known, the possible positions of the receiver can be limited to be where the two imaginary spheres intersect.

Figure 8. Two satellites [16]

Figure 9: When a <u>third</u> satellite is involved the possible positions of the receiver can be limited to two spots. It is now possible to determine a two dimensional position on the surface of earth.
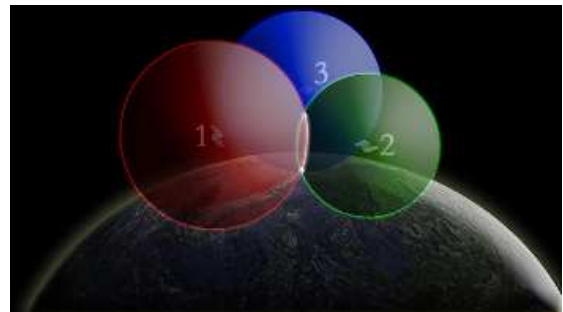

Figure 9. Three satellites [16]

Figure 10: When a fourth satellite is used the precise position on earth can be determined, i.e. a three dimensional position.

Figure 10. Four satellites [16]

6.2.5.4 Methods for determination of distance to satellites

There exist two different methods to determine the distance to the satellites.

Pseudo-range measurement

The distance to the satellites is determined through the relation $d = v \cdot t$, where d = distance, v = velocity and t = time. The signals from the satellites travel with the speed of light (v). Satellites and receivers generate common time signals. These signals make it possible to determine the time for the signal to travel from the satellite to the receiver (t). This method is almost exclusively used in absolute measurement [17].

Carrier –phase measuring

The phase measuring is based on a carrier signal. In the case of GPS there are two transmitted carrier waves, L1 and L2. Their wavelength, which is approximately 19 cm for L1 and 24 cm for L2 is used to determine the distance. This method has higher demands on the equipment and the calculations are more complex than for code measurement. Phase measuring gives a more accurate result than pseudo range measurement but takes longer to compute.

6.2.5.5 Further dividing into sub groups

When the two principles for measurement, relative and absolute measurement together with method for determine distance to the satellites, are combined with other information it is possible to divide these into subgroups. Real-time positioning or determination of position at a later time is such information.

Static measurement

Static measurement is based on relative measurement and use the carrier phase method to determine the distance to the satellites. That is, it receives differential data from a receiver on a known spot and use this to improve the accuracy. This method logs data for a period of time and determines the position at a later time and gives the highest accuracy.

Kinematic measurement

Kinematic measurement is based on relative measurement and determination of position in real time. A Kinematic measurement that uses pseudo range method to determine the distance to the satellites is called Differential GPS (DGPS). Measurements that use the carrier phase method to determine the distance to the satellites is called Real Time Kinematic (RTK).

6.2.6 Accuracy

The U.S. interfered the GPS signal for civilian until the May 1, 2000, called Selected Availability (SA) and led to a best accuracy just below 100 m. Today when the SA has been turned off, the accuracy is around 6-15 meters. Differencing GPS (DGPS), described in section 6.2.5.5 further improves the position accuracy. With the DGPS it is possible to achieve accuracy precision up to a centimetre.

6.2.7 Distortion

There are many distortions that can affect the accuracy in a GPS system. One distortion is when the satellite radio signal travels trough the troposphere and especially the ionosphere. The signals are distorted and satellites that are low on the horizon are especially exposed, since the signals travels more troughs the atmosphere compared with satellites right above.

The distortion caused by the atmosphere can be managed most of the time. The ground stations transmit correction signals to help the receiver correcting the calculation in real time.

"Multipath" is another source of distortion and can be described when additional signals are received from the satellites. These situations arise when the signals are reflected against something, e.g., a building. This problem is solved by modern receivers, which ignores the additional signal and only utilize the first signal detected, since this is the signal with the shortest path from the satellites.

6.2.8 Reference system

The GPS system uses the WGS84 reference system[7], which provides a good match between the ellipsoid and the earth's surface over the entire world. The used Datum should correspond to the one used in the map where the position is presented [18, 19, 20, 21].

---

[7] For more information on reference system see Section 7.3

6.3 GLONASS

6.3.1 Overview

GLONASS (Globalnaya Navigatsionnaya Sputnikovaya Sistema) is a positioning system developed by Russia. From the beginning it was, like GPS, a military project. It started in 1982 and was supposed to have 24 (21 + 3 spare) satellites orbiting the earth. But due to economic problems only ten are active today (Feb. 2005). During a period in 1996 all 24 satellites were in use. The satellites orbit the earth in 3 different planes every 11h and 15,73 minutes. There are two types of signals in GLONASS, standard precision navigation signal (SP) and High precision navigation signal (HP). SP is available for civil users and the horizontal accuracy is 57-70 meters and the vertical is 70 meters. If GLONASS would contain all satellites that were attended it should cover the Swedish geographic location better than the GPS, since the inclination for GLONASS is 64,8 degrees, which is in height with Skellefteå [22, 23]. GLONASS use a reference system called PZ-90.

6.4 Galileo

6.4.1 Overview

Galileo is the European counterpart to GPS. It is a global navigation satellite system that is under development and driven as a partnership between the European Space Agency (ESA) and the European Commission. It is the first satellite navigation system that has been developed for civilian and commercial use solely. But why develop a new system when GPS is available? One of the premier reasons for this is to decrease the dependence of the GPS system, since the Americans can activate the SA whenever they want. Another reason is that the responsible for GPS and GLONASS not give any guaranties about maintenance and availability of the systems. Galileo is expected to be ready in 2008 and the first satellite Giove A, was successfully launched on the 28: Th of December 2005. Galileo will be inter-operable with GPS and GLONASS, which is positive since the user will be able to determine their position with help from any satellite in any combination. The accuracy in the system will be very high and precision down to a metre will be possible for public use. High security is prioritised and the availability is guaranteed with the exception of extreme situations or circumstances. Users will be informed within seconds in case of failure of any satellites in the system. This makes Galileo suitable for safety critical applications. When the system is fully developed, 30 satellites will orbit the earth (27 + 3 spare). The inclination will be 56 degrees and the signals will give a good coverage up to North Cape [24, 25].

6.5 NMEA 0183

NMEA 0183 is an industry standard from National Marine Electronics Association. The standard was released in 1983 and defines an electrical interface and a data protocol for communication between marine instrumentation. Today is NMEA 0183 also used in non-marine instrumentation, where land navigation with GPS is one example. Most of the GPS receivers today support the NMEA 0183 standard and the protocol is one of the most commonly used. The latest version was released in January 2002. The interface or data protocol allows one "talker" and many "listener". All data transmitted with NMEA 0183 are represented by "sentences".

There are three types of sentences used in the NMEA standard, proprietary-, talker- and query sentences. Each of them starts with a dollar sign ($) and ends with <CR><LF>. However, the content and use differs.

6.5.1 Proprietary sentences

The NMEA standard makes it possible for manufactures to define their own sentences. A proprietary sentence is denoted with the letter 'P' and is followed by a three-letter manufacture ID.

In general cases the sentence can be written as:

$Psss, d1, d2, d3,...,dn, *cc<CR><LF> where:

- P for proprietary sentence
- sss is the "Sentence ID"
- d1-dn is the data fields.
- *cc is the checksum

6.5.2 Talker sentences

The five characters following the dollar sign consist of a two-letter "talker ID" and a three-letter "sentence ID". Comma-delimited data fields make up the rest of the sentence and ends up with a checksum after the asterisks "*". A sentence may contain up to 82 characters including the start and end characters.

In general cases the sentence can be written as:

$ttsss, d1, d2, d3,...,dn, *cc<CR><LF> where:

- tt is the "talker ID"

- sss is the "Sentence ID"
- d1-dn is the data fields.
- *cc is the checksum

6.5.3 Query sentences

The listeners use this sentence to request a specific sentence from a talker.

In general cases the sentence can be written as:

$ttllQsss, <CR><LF> where:

- tt is the "talker ID"
- ll is the "talker ID" of the talker being queried
- Q define that the message is a query
- sss is the "Sentence ID" requested

6.5.4 GPS and NMEA 0183

The GPS receiver work as talker and transmit the NMEA sentences to the listener, in this case the application. The "talker ID" in the talker sentences is denoted with GP, for GPS. There are about fifty different commands or sentence IDs for GPS today. Three of them, RMC, GGA and GLL, contain facts about the positions.

6.5.4.1 Recommended minimum specific GPS/Transit data (RCM)

$GPRMC, d1, d2, d3,...,dn, *cc<CR><LF> Where the data fields (d1-dn) for RCM is described below (Table 2).

Field	Description	Format
1.	Universal Time Coordinated (UTC) (Greenwich time)	HHMMSS.SSS
2.	Status: Valid/Not valid	A/V
3.	Latitude	DDMMM.MMM
4.	N/S indicator: Hemisphere (North/South)	N/S
5.	Longitude	DDDMM.MMM
6.	E/W indicator: Hemisphere (East/West)	E/W
7.	Speed over ground (knot)	D.DDD
8.	Course over ground (degree)	DDD.D
9.	Date	DDMMYY
10.	Magnetic variation (degrees)	DDD.D
11.	Magnetic variation (East/West)	E/W

Table 2. The RMC sentence

6.5.4.2 Global Positioning System Fix data (GGA)

$GPGGA, d1, d2, d3,...,dn, *cc<CR><LF> Where the data fields (d1-dn) for GGA is described below (Table 3).

Field	Description	Format
1.	Universal Time Coordinated (UTC) (Greenwich time)	HHMMSS.SSS
2.	Latitude	DDMMM.MMM
3.	N/S indicator: Hemisphere (North/South)	N/S
4.	Longitude	DDDMM.MMM
5.	E/W indicator: Hemisphere (East/West)	E/W
6.	Position Fix	0 = Invalid, 1 = Valid, 2 = Valid DGPS, 3 = Valid PPS
7.	Satellites used	Satellites being used (0-12)
8.	HDOP	Horizontal Dilution of precision
9.	Altitude	Altitude according to WGS 84 ellipsoid
10.	Altitude units	M=Meters
11.	Geoid separation value	The difference in meters between the WGS 84 ellipsoid and the main sea level (geoid)
12.	Geoid separation unit	M=Meters
13.	DGPS Age	Age of DGPS data in seconds
14.	DGPS station ID	

Table 3. The GGA sentence

6.5.4.3 Geographic position, Latitude, Longitude (GLL)

$GPGLL, d1, d2, d3,...,dn, *cc<CR><LF> Where the data fields (d1-dn) for GLL is described below (Table 4).

Field	Description	Format
1.	Latitude	DDMMM.MMM
2.	N/S indicator: Hemisphere (North/South)	N/S
3.	Longitude	DDDMM.MMM
4.	E/W indicator: Hemisphere (East/West)	E/W
5.	Universal Time Coordinated (UTC) (Greenwich time)	HHMMSS.SSS
6.	Status: Valid/Not valid	A/V

Table 4. The GLL sentence

[26, 27, 28]

7 GEODESY

Geodesy can be defined as science concerning the size and shape of earth and how to determine positions on it.

7.1 Background

This chapter explains the fundamental terms and conditions for understanding the relation between the positions determined with GPS, i.e. the geographic or geodetic coordinates and the position presented in a planar map, referred to as grid coordinates. Three surfaces are essential in geodesy and are important to understand.

7.2 Surfaces

7.2.1 Ellipsoid

An ellipsoid can be described as the surface that arises when a sphere with floating mass is rotated around its axis (Figure 11). The shape of earth has similarities with this surface but, in contrast, earth has an irregular and complicated surface. Therefore is it hard to use earth as a mathematic model for determining positions on it. The ellipsoid works as a good approximation of earth and is an easier mathematic model due to its solid shape and smooth surface.
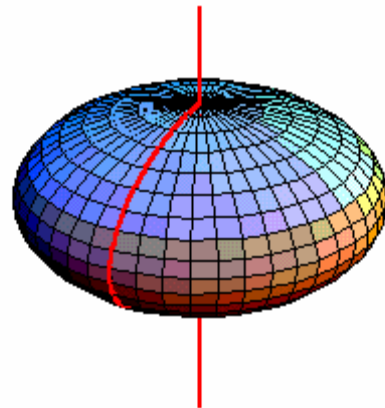

Figure 11. The ellipsoid

7.2.2 Geoid

Commonly when people speak about heights, for example, Mount Everest is located 8844 above the sea level, they actually mean 8844 above the geoid surface. The definition of a geoid can be described as a surface where the earth's gravitational forces are everywhere equal.

Further, the geoid surface can be visualised as the surface that would arise if the average sea level were allowed to float in under the continents and shape a surface that reflect the level of the average sea level (Figure 12). The surface of the Geoid coincides with the surface of oceans surface if ignoring waves and tides etc. Due to variations in gravity around the globe, the geoid surface differs irregular around the globe. The difference between the ellipsoid and the geoid surface is called the geoid height.
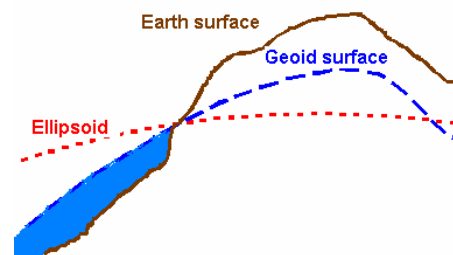

Figure 12. The geoid

7.2.3 Earth surface

The earth surface does not need any further explanation. The connections to the other surfaces are viewed in Figure 12.

7.3 Reference systems

A reference system describes the world or the area of interest, in some structured manner. As described in section 7.2.1 the ellipsoid work as an approximation of the surface of earth, with a specific size, shape and placement in relation to the geoid, the ellipsoid constitute a reference system. The parameters of the ellipsoid should be adapted so the surface of the ellipsoid coincides as much as possible with the surface of the geoid (Figure 13). A reference system is described in a geodetic datum where the size, shape and location of the ellipsoid are specified.
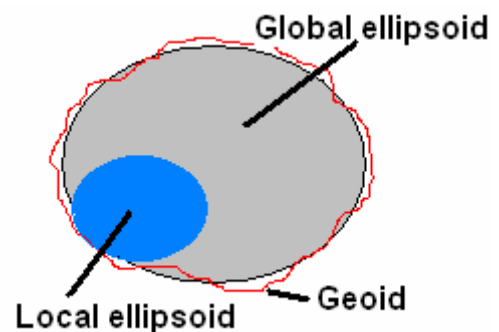

Figure 13. Reference system

A reference system also describes how we can determine positions that are related to it.

There are horizontal, vertical and geoid height reference systems, global and local that considers describing the whole world or just a limited area.

7.3.1 Global reference systems

A global reference system represents an ellipsoid that coincides with the geoid surface around the whole globe (Figure 14).

The GNSS in use today cover the whole globe and it is therefore necessary to also have a global reference system to be able to determine a position on earth.
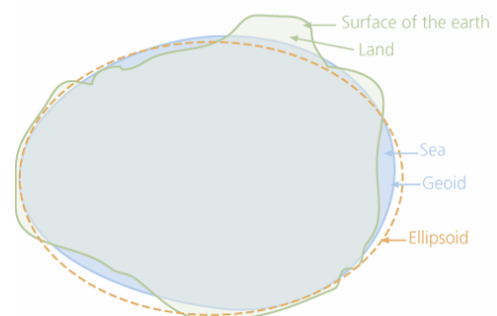

Figure 14. Global reference system

7.3.1.1 World Geodetic Datum 1984 (WGS84)

WGS84 is the most commonly used global reference system and provides an accurate fit between the ellipsoid and the geoid. This is the reference system that GPS use. GPS80 is the ellipsoid used in WGS84 reference system.

7.3.1.2 International Terrestrial Reference Frame (ITRF) and SWEREF99

ITRF, is a group of global reference systems with high accuracy.

Within Europe the EUREF is responsible for the preparations of transfer countries to ITRF. The European reference system that connects to ITRF is called ETRS89 and the coordinates are described in ETRF. EUREF come up with recommendations for nations how to join to the reference system. In Sweden has this been realized through SWEREF99, the new three-dimensional Swedish reference system.

SWEREF99 and WGS84 differ only within a couple of decimetres and can therefore be treated as the same reference system for most applications. This is of great benefit, since no conversion between reference systems is needed. For the time of this writing the transition to SWEREF99 has started, but since all data until now are saved in other reference systems there is lot of work that need to be done. New maps that refer to SWEREF99 must be delivered etc.

7.3.2 Local reference systems

A national or a local reference system fits the ellipsoid in size and position just like a global system but with the difference that it only considers the current country or area. There are many local reference systems since it is easier to find an accurate fit between an ellipsoid and a limited area of the geoid than it is for the whole globe. The global model of the ellipsoid adapt to the geoid so it differs about ±100 m. The most accurate Swedish model of the ellipsoid differs within the range ±15 m from the geoid. This is the reason why many national or local reference systems for ellipsoid are used instead of global. But this has not only been positive. Many municipalities have their own reference system and sometimes multiple reference system is used within the municipality. This complicates the cooperation between organisations that want to make use of material based on another reference system. One must the have knowledge of conversion between different systems and projections.

7.3.2.1 RT90

RT90 is one example of a local reference system. It is a Swedish horizontal reference system, which use Bessel 1841 ellipsoid. Many maps and data in Sweden are based on this system.

7.3.3 Vertical reference systems

A vertical reference system is used when heights over the ellipsoid should be transformed to heights over the geoid (height above main sea level). During the transformation is a geoid model used, which describe the surface of the geoid. The relation is depicted in Figure 15.

H = Ortho-metric height (Height over main sea level)
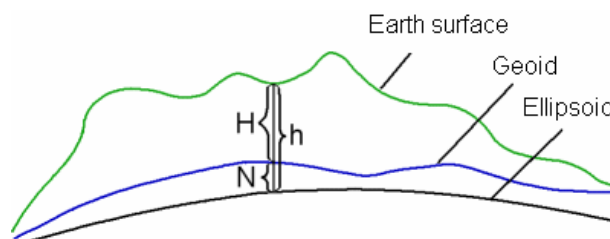N = Geoid Height (Geoid – Ellipsoid separation value)
h = Ellipsoidal height

Figure 15. Vertical reference system

7.3.3.1 RH2000

SWEN 05LR is a national (Swedish) geoid model that is used in the vertical reference system RH2000, to convert the height over the ellipsoid, received from SWEREF99 or WGS84, to heights over main sea level.

7.4 Coordinate systems

To be able to determine a position in a system one must refer the position to some sort of coordinate system. There are two different types used when a position should be determined on earth or actually the ellipsoid. Those are geodetic and geocentric coordinate systems. In a map is a two-dimensional XY-system used.

7.4.1 Geodetic coordinate system

Geographic or geodetic coordinates and are presented as latitude (φ), longitude (λ) and height over the ellipsoid (h).
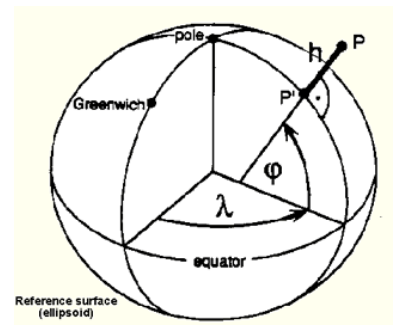
7.4.1.1 Geodetic latitude

Figure 16. Geodetic coordinate system [29]

Latitude (φ) corresponds to the angle that arises between the normal to the ellipsoid of the current spot and the equator plane (Figure 16). Latitude stretch within the range 0-90 degrees, north or south. Note that the normal from the spot doesn't intersect with the centre of the equator plane apart from the case when the normal is drawn from any of the poles or along the equator line.

7.4.1.2 Geodetic longitude

Longitude corresponds to the angle between the meridian plane and the current spot. The meridian plane can be explained as the plane that cuts through Greenwich (Figure 16) in north – south direction. A meridian is a line that stretches from North to South Pole. The zero meridians go through Greenwich and longitude stretch from 0 to 180 degree, W (West) and E (East) of Greenwich.

7.4.1.3 Height over the ellipsoid

The height in meter over the current ellipsoid, h, is not the same as height above main sea level. The geoid height is described in a vertical reference system (see sub section 7.3.3).

7.4.2 Geocentric coordinate system

A geocentric coordinate system has its origin at the mass centre of earth or ellipsoid. The geocentric coordinates are presented by the axis X, Y and Z, where the X and Y-axis is in the equator plane. The X-axis coincides with the meridian line thought Greenwich and the Z-axis with the rotation-axis of earth (
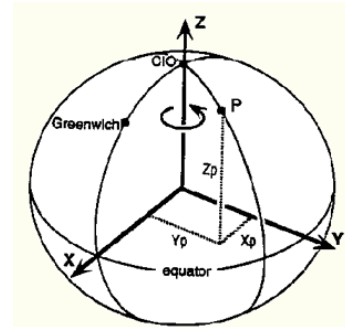
Figure 17).

Figure 17. Geocentric coordinate system [29]

7.4.3 Transformation from geodetic to geocentric coordinates

It can sometimes be necessary to transform coordinates from geodetic coordinate system to a geocentric Cartesian system, e.g. is this one of the step when one should transform coordinates from one reference system to another. The following expression shows the relation between the two coordinate systems:

$$X = (N'+h)\cos\varphi\cos\lambda$$
$$Y = (N'+h)\cos\varphi\sin\lambda$$
$$Z = (N'(1-e^2) + h)\sin\varphi$$

$$N' = \frac{a}{\sqrt{1-e^2\sin^2\varphi}}$$
$$e^2 = f(2-f)$$
$$f = \left(\frac{a-b}{a}\right)$$

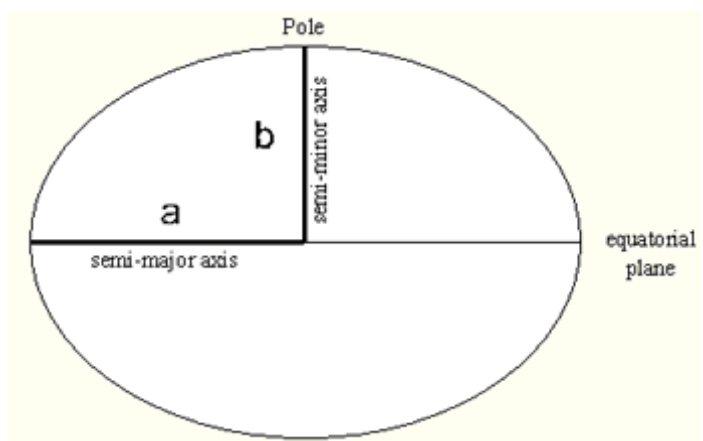**Figure 18. Semi-major and semi-minor axis [29]**

Where

(X, Y, Z)	= Geocentric coordinates
(φ, λ, h)	= Geodetic coordinates
N´	= Radius of curvature in the prime vertical
e^2	= First eccentricity squared
f	= Flattening
a	= Semi major axis

The inverse e.g. transformation from geocentric to geodetic coordinates follows the relation:

$$\lambda = \tan^{-1}\left(\frac{Y}{X}\right)$$

$$\varphi = \tan^{-1}\left(\frac{Z + \left(\frac{ae^2}{\sqrt{1-e^2}}\right)}{p - ae^2 \cos^3 \theta}\right)$$

$$h = \frac{p}{\cos \varphi} - N´$$

Where
$$p = \sqrt{X^2 + Y^2} \qquad \tan \theta = \frac{Z}{p\sqrt{1-e^2}}$$

7.4.4 Plane coordinate system

A two-dimensional coordinate system is used to refer to positions in a flat map. The positions are described with X and Y coordinates or as Northing and Easting. They are also referred to as Cartesian coordinates. One should pay some extra attention to the fact that Northing is expressed as X and Easting as Y, logically it should be the opposite in an ordinary X-Y coordinate system.
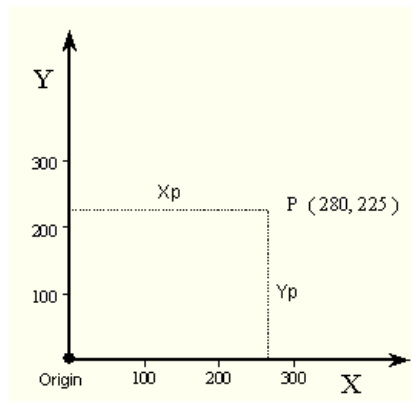

Figure 19. x-y coordinate system [29]

7.5 Conversion between different reference systems

The general way when converting from one reference system to another (different geodetic datum) is through the 7-parameter formula. One disadvantage with this method is that vertical information must be added even though it is only plane coordinates that should be converted. The formula use geocentric coordinates.

7.5.1 The 7 parameter formula

The formula performs three translations, one for each axis, three rotations, one around each axis and one scaling.

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{To} = \begin{pmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{pmatrix} + (1+\delta)R\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{From}$$

Where:

- $\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{To}$ is the geocentric coordinates in the new reference system.

- $\begin{pmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{pmatrix}$ is the translation between the two systems (Figure 20).
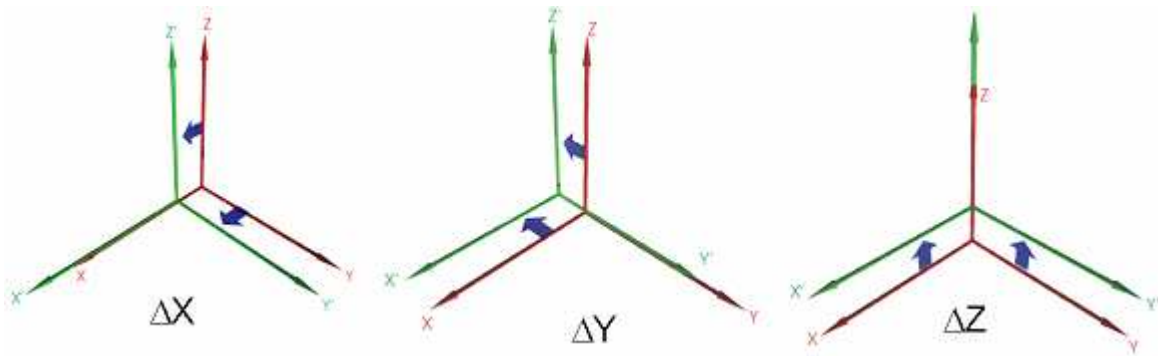

Figure 20. Translation for each axis

- $R = R_Z R_Y R_X = \begin{pmatrix} \cos\omega_Z & \sin\omega_Z & 0 \\ -\sin\omega_Z & \cos\omega_Z & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\omega_Y & 0 & -\sin\omega_Z \\ 0 & 1 & 0 \\ \sin\omega_Y & 0 & \cos\omega_Y \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\omega_X & \sin\omega_X \\ 0 & -\sin\omega_X & \cos\omega_X \end{pmatrix}$

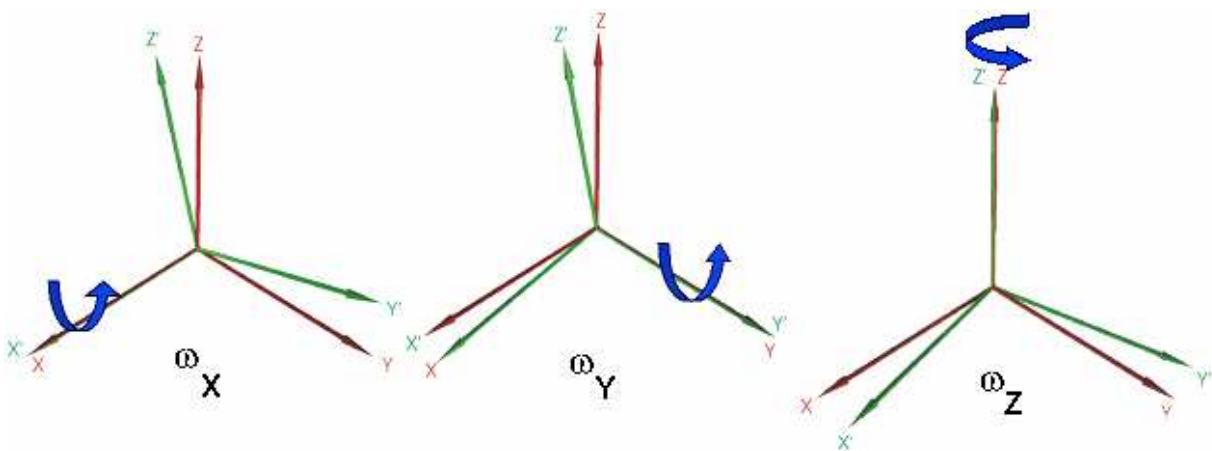**Figure 21. Rotation around each axis**

- δ is the scale factor.

- $\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{From}$ is the geocentric coordinates in the "old" reference system.

The inverse follows this expression

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{To} = \frac{R^{-1}}{((1+\delta))} \left[\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}_{From} - \begin{pmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{pmatrix} \right]$$

7.5.1.1 Parameters for conversion from WGS84 to RT90

$\Delta X = -414.0978567149$ $\omega_X = -0.8550434314$

$\Delta Y = -41.3381489658$ $\omega_Y = +2.1413465185$

$\Delta Z = -603.0627177516$ $\omega_Z = -7.0227209516$

$\delta = 0.0$ ppm

7.5.2 "Direkt projektion"

At Swedish lantmäteriet has a method been developed that transform the coordinates from latitude, longitude in one reference system directly into grid coordinates in another reference system. The method is called "Direkt projektion" and it use a projection called Gauss Kruger projection. The whole process is handled in one step unlike the 7 parameter formula. The two methods result in about the same accuracy.

7.6 How position are presented

The international standard ISO 6709-1983 state how the position should be presented:

Latitude:

Degrees and decimal degrees: DD,DD
Degrees, minutes and decimal minutes DDMM,MMM
Degrees, minutes, seconds and decimal seconds DDMMSS,SS

Where:

One degree is equal to 1/360 of an orbit.
One minute is equal to 1/60 of a degree.

One second is equal to 1/60 of a minute.

Northerly latitude is described with "+" and southerly as "-"

The number of decimals in decimals -degrees, minutes and seconds can be extended to increase the accuracy.

Longitude:

Longitude follows the same pattern as for latitude with one exception. The degrees are presented with three numerals, this since longitude stretch from 0-180 degrees, west and east and latitude are presented within the range 0-90 degrees north and south.

Easterly is described with "+" and westerly with "-"

Height over ellipsoid:

Described as positive "+", or negative "-" altitude in meters over the ellipsoid [30, 31].

7.7 Map projections

This chapter has so far explained that there are different kind of reference systems and how it is possible to convert from one reference system to another. The purpose with this chapter was to show the relation between the coordinates received from the GPS and the one presented in a planar map. The map projection can be seen as the last step in this process.
Where the geographic coordinates, latitude, longitude and h, is projected into plane (grid) coordinates, x- and y coordinates, or Easting and Northing. There are three main types of map projections, cylindrical, azimuthal and conic.

7.7.1 Cylindrical projection

The earth is projected on a cylinder that is wrapped around the world. The cylinder can then be unrolled into a plane (Figure 22). There are two main types of cylindrical projection, normal cylindrical projection and traversal cylindric projection. These two models differ in the way earth axis corresponds to the symmetry axis of the cylinder. In the normal cylindrical projection the rotation axis of earth coincide with symmetry axis of the cylinder and in traversal cylindric projection the rotation axis of earth is at right angles to the symmetry axis.

Figure 22. Cylindrical projection [32]

7.7.1.1 Gauss Krügers projection

Swedish maps based on RT90 use a traversal cylindrical projection called Gauss Kruger projection.

7.7.2 Azimuthal projection

A plane is located so it touches a specific spot on the "globe". The earth is then projected on the plane, with a projection centre in the centre of earth or the nadir point (Figure 23).
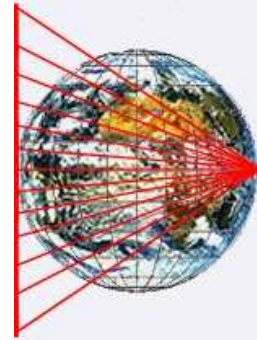

Figure 23. Azimuthal projection [32]

7.7.3 Conic projection

The earth is projected on a cone that surrounds the earth. The cone can then be unrolled to a plane (Figure 24).

Figure 24. Conic projection [32]

Student	Doc Nr		Säk klass
Markus Pettersson			
Supervisors on CC-System	Date	Rev	Filename
Johan Persson and Jörgen Martinsson	2006-05-15		GIS platforms for WinCE

CC SYSTEMS

7.8 The process from WGS84 to Grid coordinates in RT90

In this chapter have each step in the process from geographic coordinates to grid coordinates now been described. Figure 25 was made to clarify what support for a Swedish reference system (RT90) actually means in our case. I.e. convert the position received in WGS84 to geocentric coordinates and then transform the position through the 7-parameter formula to the Swedish reference system RT90. Since the position in RT90 now is described in a geocentric coordinate system a conversion to a geodetic coordinate system is made. Finally a projection trough the Gauss Krügers projection is performed. A conversion to other reference systems and the use of other projections follows the same pattern [21, 29, 33, 36].
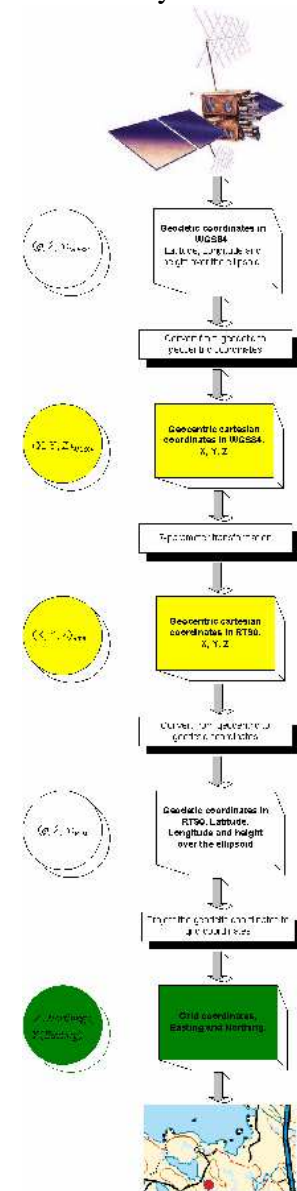

Figure 25. From geodetic to grid coordinates

8 SPATIAL DATA MODELS

There are two ways to handle geographic information in a GIS. The models are raster- and the vector spatial data model.

8.1 Vector data

The vector format uses Cartesian coordinates (x, y) to represent the different shapes, to with attributes can be connected. The different shapes are:

- Points (described with a single coordinate, Figure 26)
- Line (Two or more coordinates linked to a sequence, Figure 27)
- Polygon or Area (a sequence of lines where the first and last coordinates are the same, Figure 28)
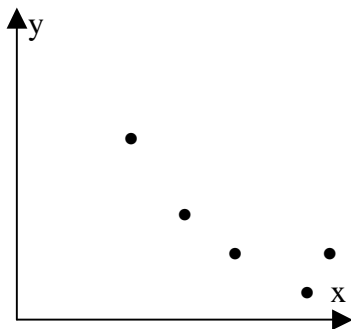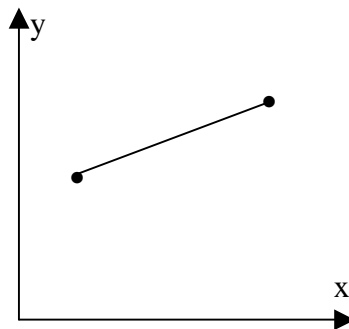
Figure 26. Points in vector format

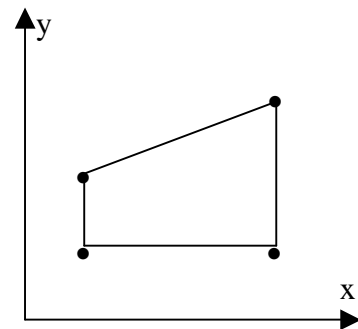**Figure 27. Line in vector format**

Figure 28. Polygon in vector format

8.2 Raster data

The raster format consists of rows and columns of cells, where each grid holds a geographic location. The most common use of the raster format is for images, but the grids can also hold other information than colour, e.g. precipitation for a specific geographic location.
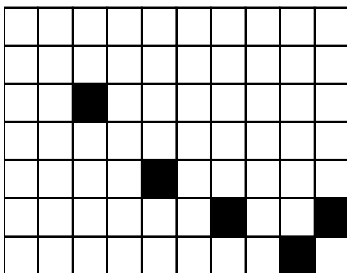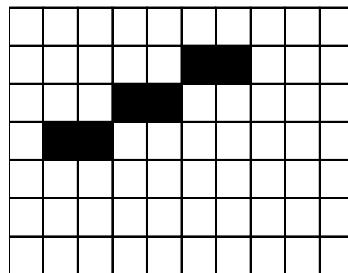
Figure 29. points in raster format
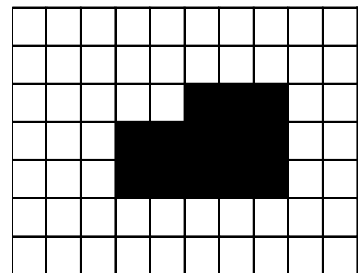
Figure 30. Line in raster format

Figure 31. Polygon in raster format

8.3 ESRI vector data format

One popular and common vector format is ESRI shapefiles. The structure of the shapefile includes three different types of files that together create a dataset. The file with the suffix .SHP handle the geometry (x,y), the suffix .DBF contains possible attributes to the shapes in the .SHP file. The last file, with the extension .SHX, connects the two former. The shapes described in subsection vector data are often referred to as Geoobjects in a GIS.

8.4 Layers

The most common method to structure the geography is to use a layered approach, where each layer describes a theme, e.g. lakes and roads (Figure 32). This is possible with both of the spatial data formats. A layered structure can be very helpful for the user of a GIS since he/she can view the layers that are interesting for the moment, and by those means exclude irrelevant information [35, 36].
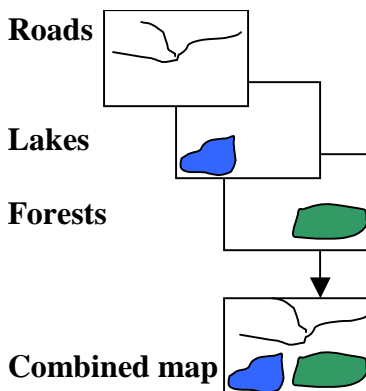

Figure 32. Layered approach

9 COMPARISON BETWEEN EXISTING SOFTWARE

To choose appropriate software or core adaptable to the customer's demands, some alternatives will be presented in this chapter. These alternatives are just a fraction, but the ones we today found most appropriate.

9.1 SpatialAce Light Edition (LE)

SpatialAce LE from Carmenta, is a toolkit for building wireless GIS applications. The product is based on its big brother SpatialAce, a desktop variant of the toolkit suit. SpatialAce LE is optimised for use on simpler hardware platforms, such as handheld computers, PDAs and cell phones. SpatialAce LE can be used to develop applications for devices using Java Micro Edition or .NET Compact Framework. Further it can be used on ruggedized or vehicle mounted computers using java standard edition or .NET. The toolkit can use a combination of local GIS and map layers that are accessed via a remote server. SpatialAce has no built in functionality to access GPS data [37, 38, 39, 40].

9.2 MapX Mobile

MapInfo is one of the leading companies within GIS today. MapInfo MapX mobile v.5.0 is a development tool for Pocket PC developers. With MapX it is possible to create customized applications via standard development languages such as Embedded C++ and .NET. Applications built on MapX Mobile do not require wireless connectivity. The application and data reside right on the device. Applications can also connect to Microsoft SQL server 2000 CE edition via ADO [41, 42, 43].

9.3 ArcPad

ArcPad is a software for mobile mapping and GIS. Behind the product is one of the most established companies within GIS today, Environmental System Research Institute (ESRI). ArcPad provides database access, mapping, and GPS integration to users with handheld devices. Unlike the other software mentioned above, ArcPad is a software ready to run. It can be adapted to customers need through a tool called Application Builder. Since this software is a product ready to run, we know that it could be used to fulfil the demands of our application. But it is not only the functionality that affects the choice of software or core to be adopted. It also has to be easy to customize and it should be possible to adapt it to further needs. The Graphical User Interface (GUI) should also be possible to affect to get a personal touch. As mentioned earlier, ArcPad is adaptable but not as flexible as you could wish [44, 45, 46].

9.4 GpsTools

GpsTools is a product developed by Franson Technology. Franson offers solutions for the GPS software market. GpsTools is an easy and cost effective way to develop GPS mapping and basic GIS applications in Visual Studio. With help from GpsTools it is easy to access GPS position, speed and satellite information. There is support for vector and raster data. It is also possible to convert coordinates between different coordinates systems, something that is necessary to receive planar coordinates from the geographic coordinates received from the GPS. It is possible to rotate and zoom the map. GPSGate is an additional tool from Franson Technology. With GpsGate it is possible to simulate a GPS device during development [47].

9.5 MapACE DLL

Graticule is a company within Digital mapping. One of their products is a software component library for mapping in Windows CE environment. To develop applications using the MapACE DLL eVC+++ 4.0 (or Visual studio 2005) is used, i.e. only native code development is possible. There is support for map and raster data, other function possible to implement with MapACE are panning and zooming, rotate vector map data, handling layers etc [48].

9.6 MapClassLib.Net

MapClassLib.Net is a class library for mapping and is also developed by Graticule. It targets .NET runtimes environments, including Windows CE 5.0. The MapClassLib DLL include classes for map display. It is possible to import raster and vector data, pan, zoom and handling of layers [48].

9.7 GeoFrameworks GPS.NET and GIS.NET

GeoFrameworks is specialized in creating software components targeting the Microsoft .NET platform. Their firm product is GPS.NET, which helps the user with basic GPS functionality.

Among others, support for automatic GPS device detection and creation of own virtual GPS device is included. One Class convert raw data from NMEA strings into objects. To achieve mapping functionality GeoFrameworks offers GIS.NET. These two products, together with an ESRI Shape file plug-in, which enable the use of ESRI shape files, would work as a stable ground for a navigation application with the specified demands from CC systems. GPS.NET and GIS.NET are available for a wide range of platforms including Windows CE, Pocket PC and the Windows mobile 5.0 [49].

9.8 Comparison

A comparison between the seven alternatives in matrix form follows, and will work as a basis for the decision of appropriate core for the demonstrator.

Name	ArcPad	Map X Mobile	SpatialAce Light Edition	GpsTools	GPS.NET GIS.NET	MapACE DLL	Map ClassLib.Net
Company	ESRI	MapInfo	SpatialAce	Franson	GeoFrameworks	Graticule	Graticule
Vector data	ESRI Shapefiles (.shp)	MapInfo TAB format (.tab)	ESRI Shapefiles (.shp)	ESRI Shapefiles (.shp)	ESRI Shapefiles (.shp)	ESRI Shapefiles (.shp) DXF BNA VFF NTF MIF EMF	ESRI Shapefiles (.shp)
Raster data	MrSID JPEG BMP PNG CADRG ArcIMS	MrSID JPEG GIF BMP PNG TIFF ECW SPOT PCX Photoshop Targa WMF SPEG200	JPEG GIF BMP PNG	GIF BMP	No information of raster data	BMP WMF TIFF	JPEG GIF BMP PNG TIFF MIFF
GPS Protocol support	Built in GPS support: • NMEA • TSIP • DeLorme Earthmate binary protocol • FLGR	Built in GPS support: Applications can import and display coordinates from a GPS device	No built in GPS support.	Built in GPS support: Access to GPS position, speed and satellite information. • NMEA support	Built in GPS support.	Built in GPS support: Contains functions that help you to parse NMEA	No built in GPS support, planned however.

Name	ArcPad	MapX Mobile	SpatialAce Light Edition	GpsTools	GPSNET GISNET	MapACE DLL	MapClassLib.Net
Adaptable	Yes with ArcPad Application builder which include ArcPad studio	Yes with Microsoft Embedded tools using embedded C++ or .NET. Windows CE 2003 device and emulator support using the eVC++ 4.0 compiler, Service Pack 3.	Yes, The SpatialAce Development Environment or Visual studio.NET. SpatialAce LE is implemented in Java 2 Standard Edition (J2SE), Java 2 Micro Edition (J2ME), NET and NET CF	Yes, Visual Studio 2003 pro or Visual Studio 2005 standard or pro.	Yes, with Visual Studio 2002, 2003, 2005	Yes, eVC++ 4.0, Visual Basic (Win CE 3.0 only)	Yes, any .NET, e.g. C#.NET, Visual Basic .NET, J#, Delphi, Managed C++
CPU support	- ARM(including StrongARM and XScale) - Hitachi SH3 and SH4 - MIPS - x86	- StrongArm or Xscale					
Supported OS and environment	• Win Mobile 2003 2nd edition for Pocket PC 2003 • Windows CE.net (4.1 and 4.2) • Windows CE 2.11, 2.12, 3.0 • Windows 95/98/2000, Me, NT,XP	• Pocket PC 2003 • Pocket PC 2003 phone edition.	• J2SE • J2ME • NET & NET CF • Windows Pocket PC	• Windows • PocketPC • Windows CE • ActiveX • NET (1.0, 1.1 and 2.0)	• Any with .Net or Compact Framework	• Windows CE 3.0 • Windows CE 4.2 • Windows CE 5.0	• Any with .Net or .Net compact framework

Name	ArcPad	MapX Mobile	SpatialAce Light Edition	GpsTools	GPS.NET GIS.NET	MapACE DLL	MapClassLib.Net
Price	ArcPad Builder: 21 500 SEK, 1 arcpad license included. Each adapted version of arcpad requires an ArcPad license for 6 500 SEK each.	SDK: Free License: SEK 2500/unit	Single developer license: 30 000 SEK/developer Runtime license: approx. 2000 SEK/user	Single developer OEM licence $349, royaltyfree. GpsGate $9.92 /unit	For each development machine (royalty free) GPS.NET$249 GIS.NET$299 Plug-in for ESRI shapefiles $149 Additional plug-in $179+$99	Single developer license: 495£, runtime licence (min.25): 250£	Single developer license: 995 £, runtime licence: call
Demo version	Yes, unlimited time, Application restart after 20 minutes	Yes, 30 days.	Yes, 30 days, possible to get expanded demo time during project	Yes, 30 days, GpsGate 14 days	Yes, one month	No	No

10 DISCUSSION AND CONCLUSIONS (SURVEY)

The goal was to find a suitable component for GIS. The one that fulfilled our needs was Franson GPS Tools for .Net compact framework. The product had support for Bmp and GIF raster images and also ESRI shapefiles. It was possible to convert a position between different reference systems and the position could also be projected to grid coordinates. The component supported the Swedish reference system RT90. It simplified the connection between the application and the GPS receiver and supported the NMEA 0183 protocol.

	Student Markus Pettersson	Doc Nr	Säk klass
	Supervisors on CC-System Johan Persson and Jörgen Martinsson	Date 2006-05-15	Rev	Filename GIS platforms for WinCE

47 (76)

11 DEMONSTRATOR XS NAVI

11.1 The choice

Fransons GpsTools has support for GIF and BMP raster images and also support for ESRI shapefiles. This makes it possible to view a map. The product simplifies the connection between the application and the GPS receiver, contains classes that parse NMEA 0183 sentences, support the Swedish reference system RT90, i.e. the position can be converted into the correct map Datum and projected into grid coordinates. These features make it possible to show the current position on the map. The position can then be visualized by drawing a mark for the current GPS position.

There was also some desirable functionality, which would be handled if it were possible and if time allowed. One of those was layer handling. This functionality has been implemented and it is possible to add and remove layers. Enable and disable layers. Track recording is another thing that was possible to implement. Using Fransons GpsTools support for shapefiles has solved this. Support for waypoints can easy be implemented using a class method that calculates the bearing between two coordinates. The former have been left out as future work.

Additionally it is possible to target the .NET Compact Framework and implement the demonstrator in C#. After the survey of development options this had appeared to be the best and most powerful alternative. Fransons GpsTools also has a searchable forum for questions concerning the product. Good guidance in form of examples, fast support and advantageous prices.

11.2 Franson GpsTools

Franson GpsTools for .NET Compact Framework is a class library delivered as three dll: s:

- GpsToolsNET
- GpsViewNET
- GpsShapeNET

GpsToolsNET contains classes helping developers with GPS programming, i.e. there are classes for parsing of NMEA0183 sentences. One of the classes handles the serial connection to the GPS receiver. There is support for conversion between different map datum and projection into grid coordinates. It is also possible to define own Datum and Grids (projection to grid coordinates).

GpsViewNET contain classes for control of the map view, classes for displaying raster maps, drawing shapes in the map view and handle layers.

GpsShapeNET contains two classes, one for creating shapes and one handling shapefiles.

11.3 The design

The original design in Figure 33 for the demonstrator is based on the required features in the beginning of the project. The design has later, for the time of implementation, been extended and refined to what finally has been implemented. This is presented in Figure 34 on the next page.
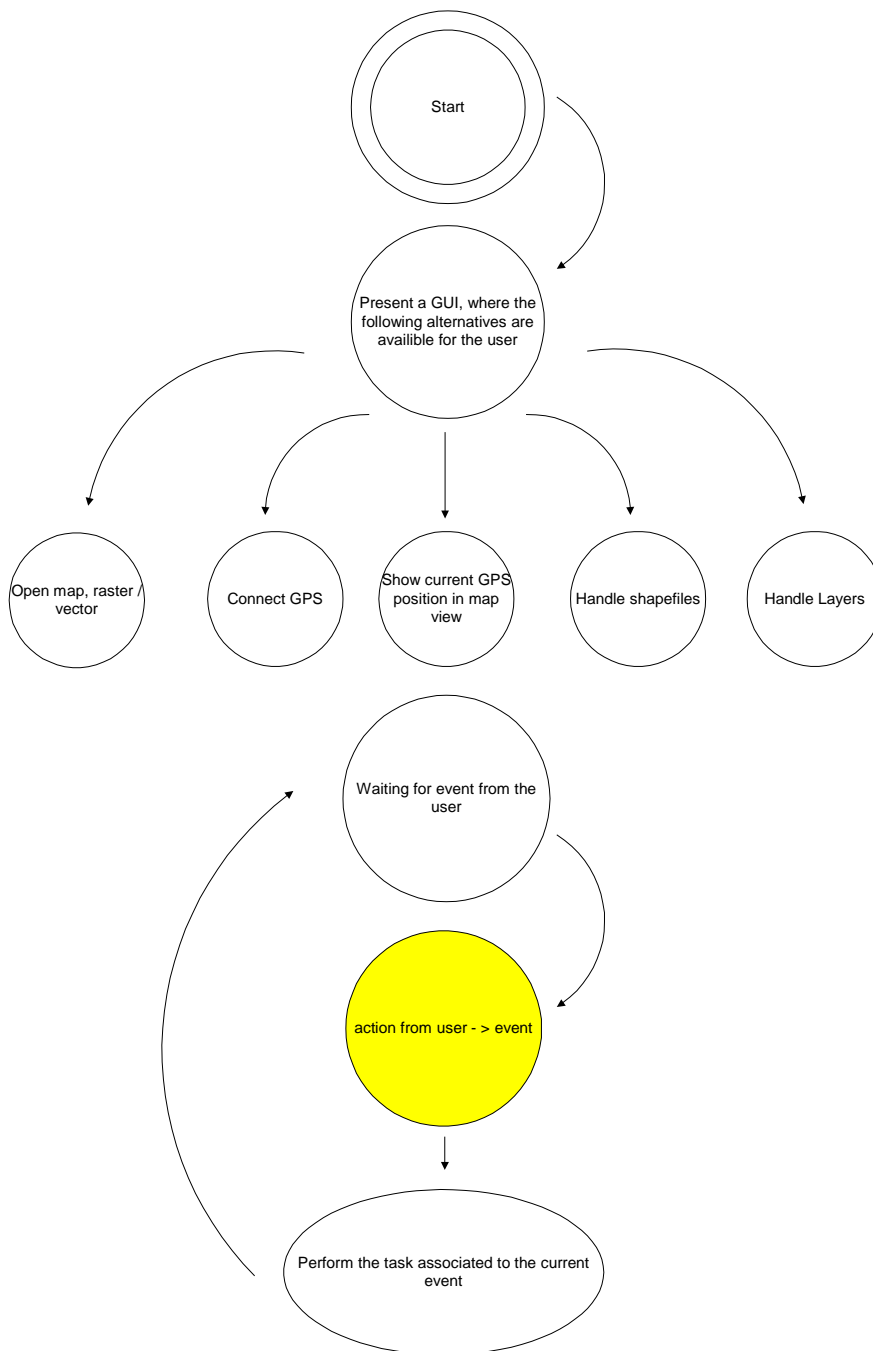

Figure 33. The original design
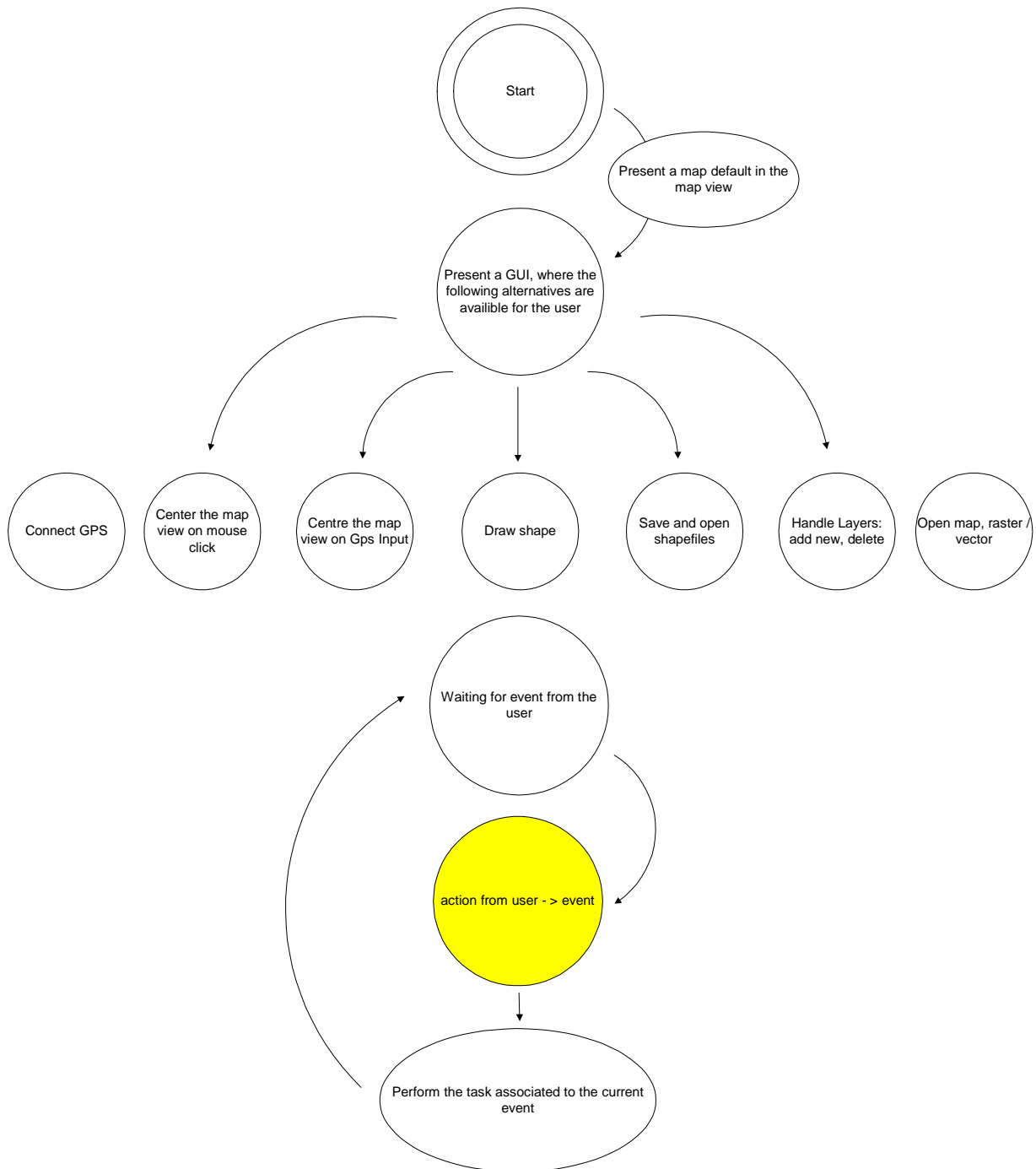
11.4 Final design

Figure 34. The final design

11.4.1 Open raster map

Before a raster map in GIF or BMP can be displayed in the map view it must be defined to a proprietary format (MapLib). This is done in GpsTools Studio (see Appendix c). The raster image can either be opened as a background image or merged into an already existing background image. A class called Map handle the map view in GpsTools. This class has been used during implementation to open a map. In the demonstrator is it only possible to open the raster maps as background images.

11.4.2 Vector maps

11.4.4.1 Open vector map as background

There is no class method in GpsTools for opening a vector map as a background image. This is solved by first create a blank raster map (MabLib) and then draw the vector image on top of this blank image. There must always be a raster image as background image. The blank raster map has been scaled and adapted to the extreme values of the vector map.

11.4.4.2 Open vector map in a new layer

When a background image exists in the map view, a vector map can be drawn on the existing layer or on top as a new layer. If the vector map should be drawn in a new layer one must first add a new layer in the map view. There is a class method for this in the class Map, the class that controls the map view. The vector image is then drawn on the current layer, shape by shape (Figure 35).
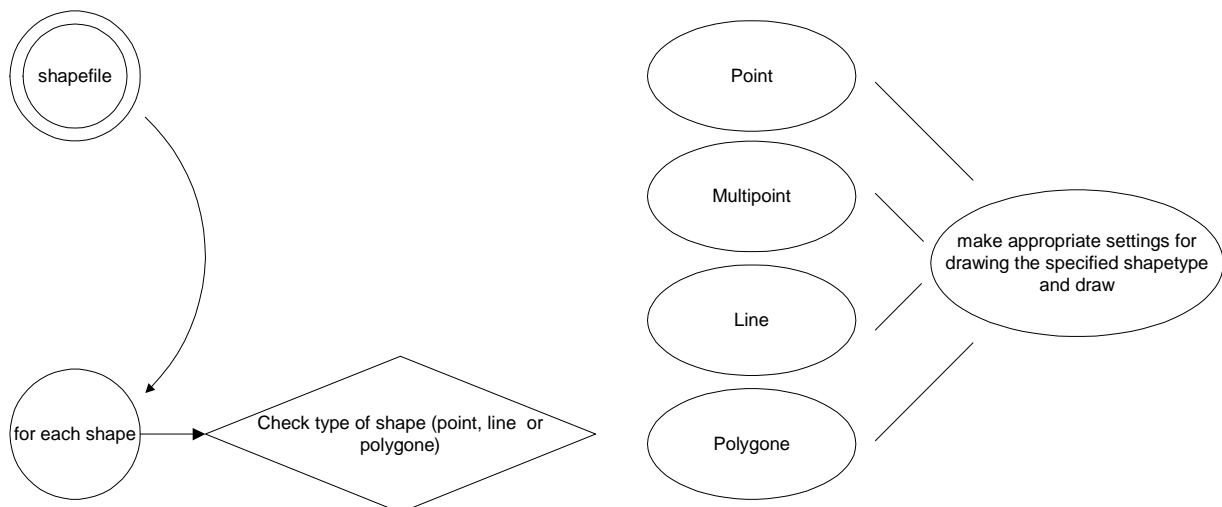
Figure 35. Draw vector map in new layer

11.4.3 The GPS functionality

Franson GpsTools and the class NmeaParser handle the establishment of a connection between the application and the GPS receiver. When the user tries to connect to the GPS device, the search for a GPS begins. The additional functionality in the application that concerning the GPS is ruled by three events that are triggered when:

- something happens on the serial link.
- a position is determined (when a RCM, GLL or GGA sentence is received with a valid position).
- a movement is determined (a RMC sentence has been received) and the tasks that should be performed are handled with an event handler for each event.

Each of these events is caught and handled by an event handler. These are described in more detail on the following pages.

11.4.3.1 Serial link to the GPS
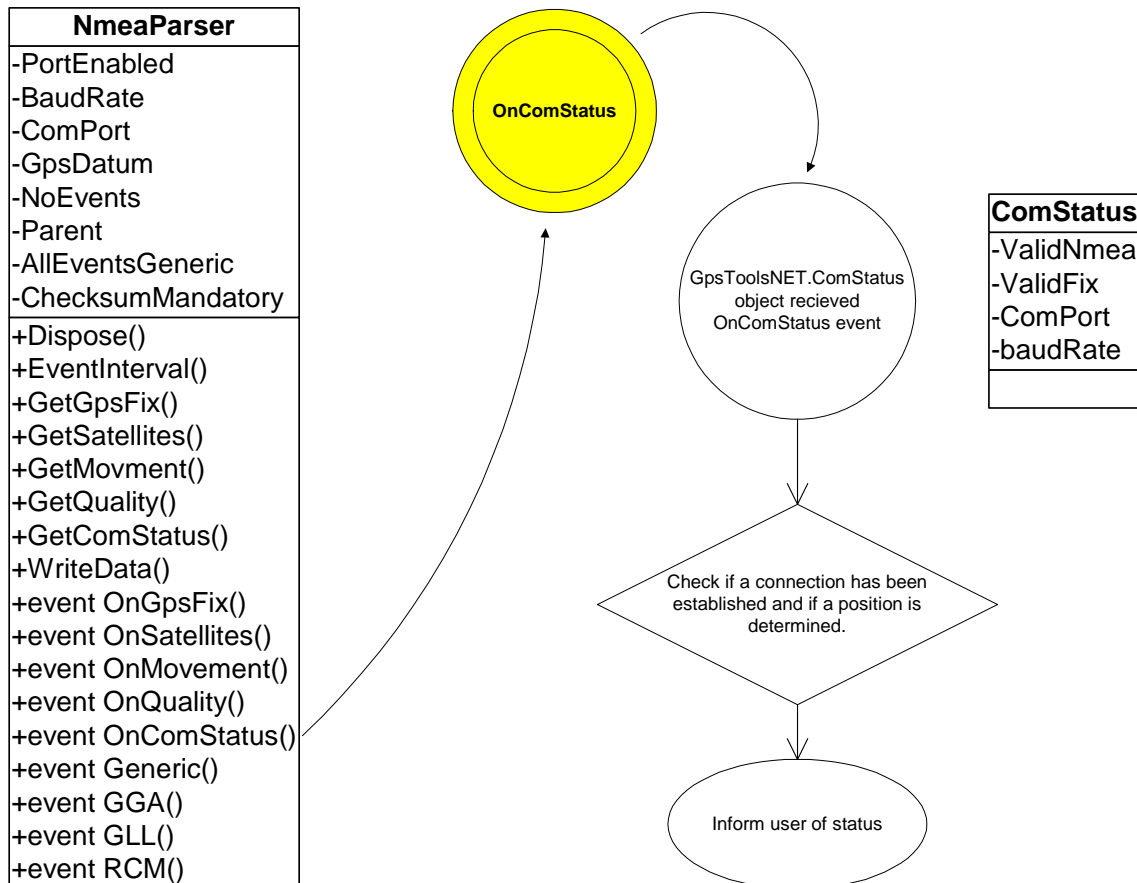
Figure 36. Description of the OnComStatus event

The OnComStatus event is generated when something happens on the serial link to the GPS. On event is a ComStatus object received. This object contain the following information:

- ValidNmea: True if a connection to the GPS has been established.
- ValidFix: True if a position is determined.
- ComPort and baudRate contains information of the connection.

This event is used to inform the user of the current GPS status (Figure 36).

	Student Markus Pettersson		Doc Nr		Säk klass
	Supervisors on CC-System Johan Persson and Jörgen Martinsson		Date 2006-05-15	Rev	Filename GIS platforms for WinCE

CC SYSTEMS
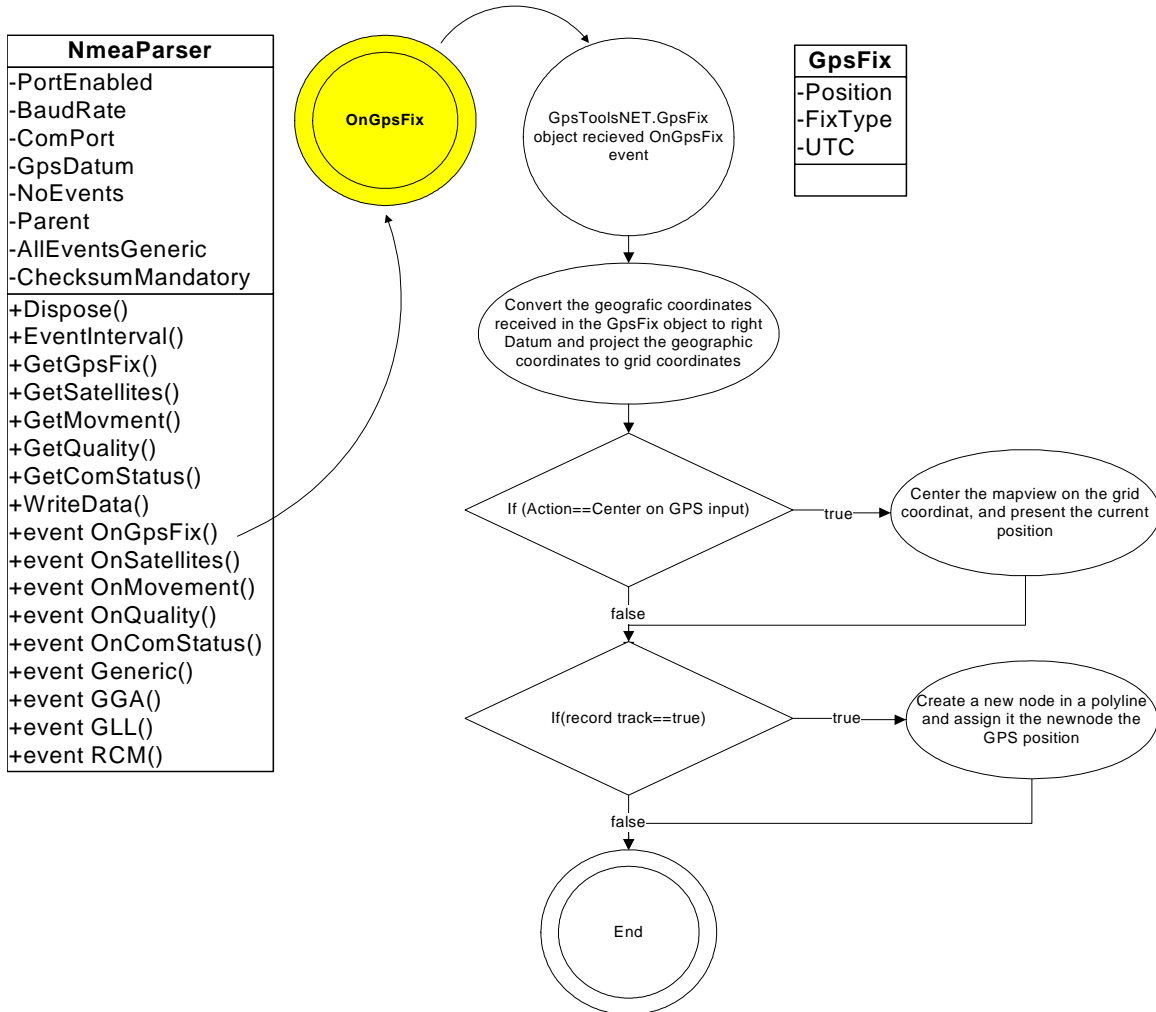
11.4.3.2 GPS Position

Figure 37. Description of the OnGpsFix event

The OnGpsFix event is generated when a valid position is received from the GPS. On event is GpsFix object received. This object contains the following information:

- Position: The current GPS position
- FixType: Informs if the position contain a two- or three-dimensional position.
- UTC: Contains the time stamp of the positioning.

When receiving this event the object has been used for centring the map view on current GPS position, track recording etc. (Figure 37).

11.4.3.3 Speed and heading
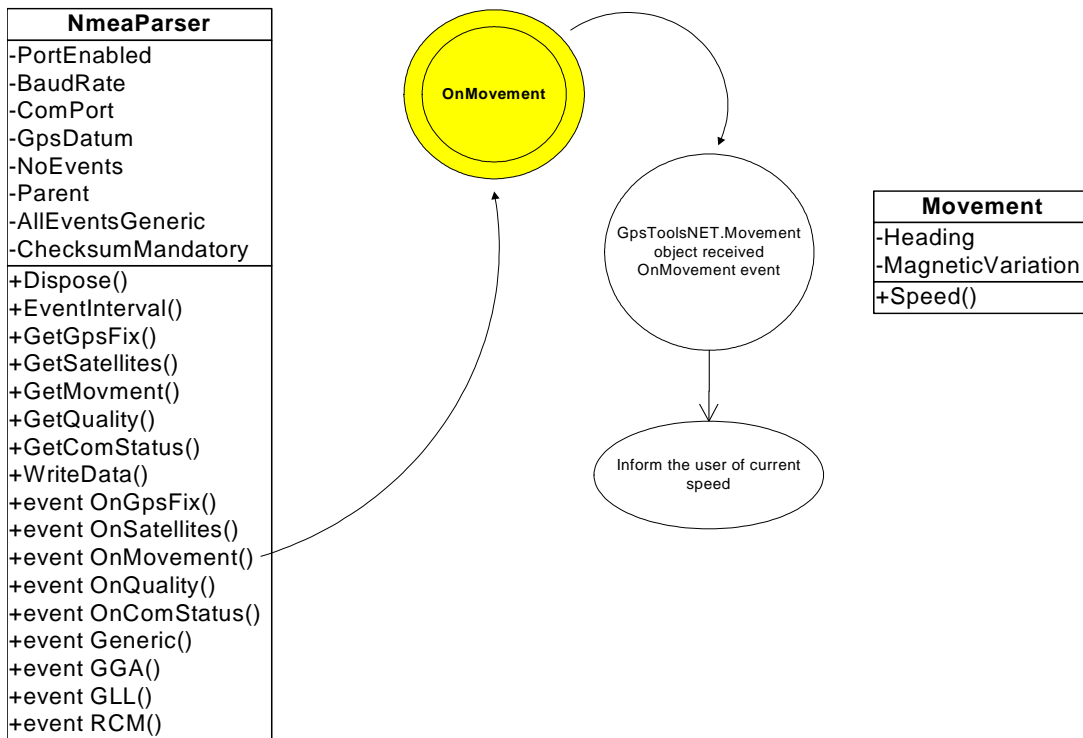
Figure 38. Description of the OnMovement event

The OnMovement event is raised when the speed and heading is determined. A Movement object is received on event. The object contains:

- Heading: The current heading in degrees
- MagneticVariation: The variation in degrees between the heading in a compass and the heading received from the Movement object. The heading received from a compass is based on the magnetic poles, while the heading received from the Movement object is based on the real poles.
- Speed: The current speed.

This event is only used to inform the user of the current speed when the GPS is connected (Figure 38).

11.4.4 Move around in the map

It must somehow be possible to move around in the map. By centring the map view on the spot the user click on solve this. The event for clicking on the map view is also used for drawing shapes. So, when the user click on the map, it can either be since he/she want to centre the map on this spot or he/she want to draw a node in a shape.
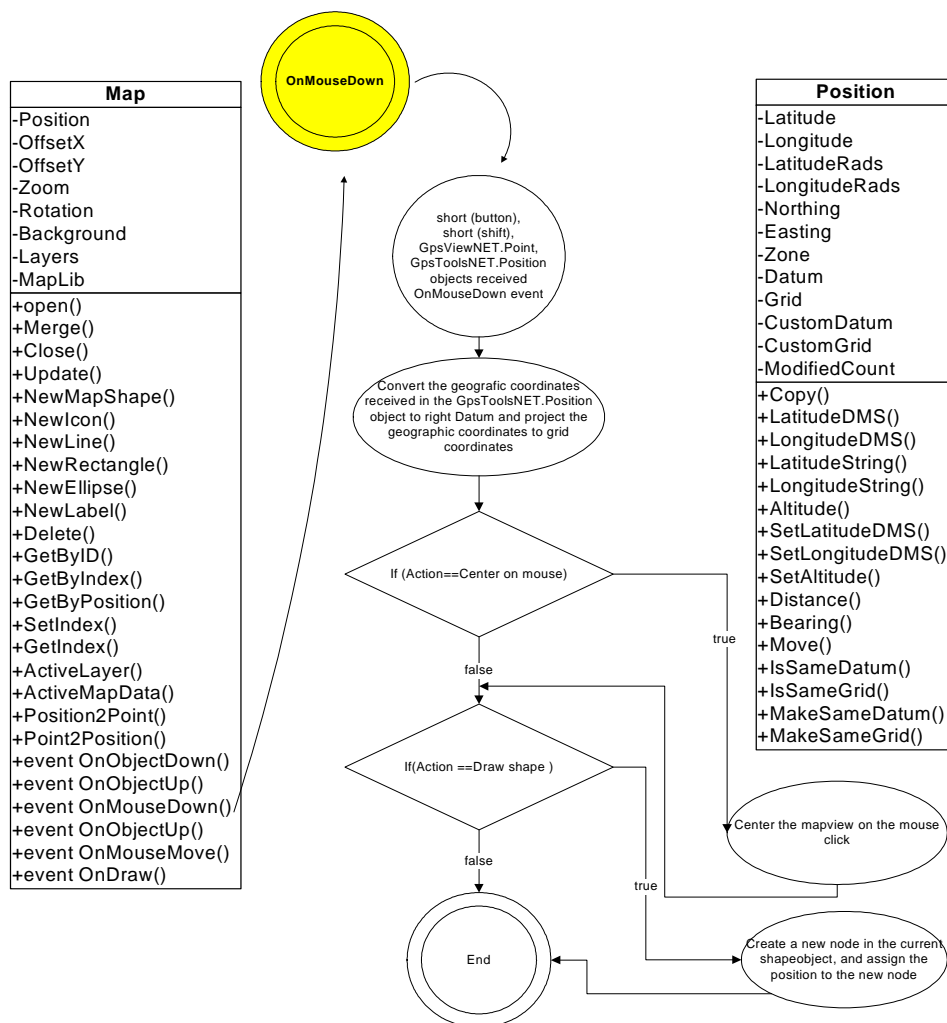
Figure 39. Description of the OnMouseDown event

The OnMouseDown event is raised when the user click on the map view. On this event is one position object, one point object, and two short (button and shift characteristics) received. The task that should be performed is handled with an event handler (Figure 39) and concern drawing and centring of map view.

11.5 The prototype
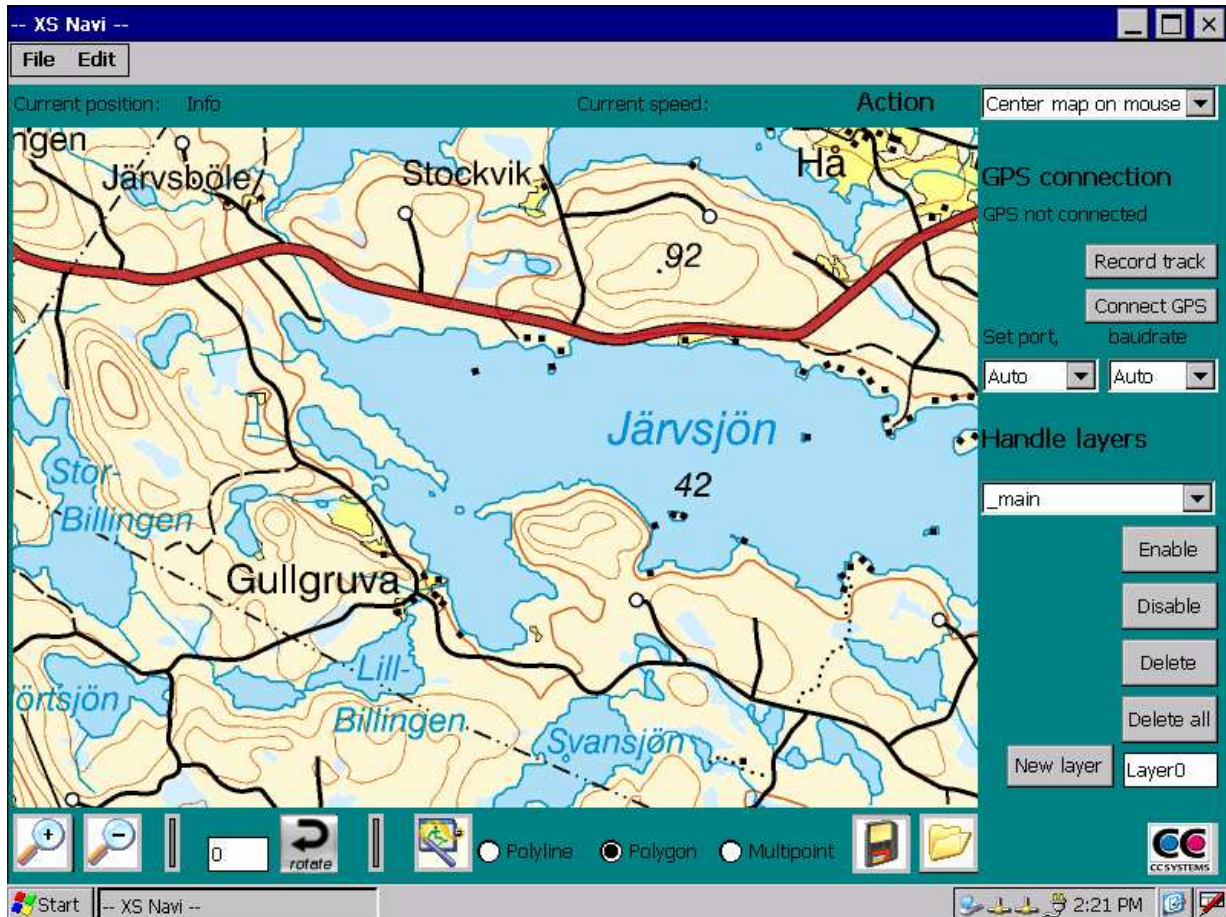
Figure 40. The result of the implementation

The demonstrator is implemented in Visual Studio 2005 and the programming language is C#.NET. The aim with the demonstrator was to show that the required features could be fulfilled with a suitable component, in this case Franson GpsTools for .Net Compact Framework.

The requirements being presented in Chapter 1.3 were divided into minimum requirements (things that must be fulfilled and implemented) and desirable functionality, things that would be fulfilled if possible and if time allowed. The demands for minimum requirements have been implemented and additionally has also layer handling, drawing own shapes that could be saved to file and track recording been implemented. These were some of the functionality stated in desirable functionality. The result of the implementation shows that the task was possible to fulfil with Franson GpsTools.

12 DISCUSSION AND CONCLUSIONS (DEMONSTRATOR)

After the choice of component the implementation of the demonstrator could start. The work with the demonstrator went well and obviously we made a good choice since the implementation of the demonstrator showed that all minimum requirements were and could be fulfilled. Additionally was several of the desirable functionality also obtained. My conclusion about this is that even if the survey were a bit delayed, it paid back when it was time for implementation.

Student		Doc Nr		Säk klass
Markus Pettersson				
Supervisors on CC-System		Date	Rev	Filename
Johan Persson and Jörgen Martinsson		2006-05-15		GIS platforms for WinCE

58 (76)

13 REFERENCES

[1]　　　　　CC Systems, Exjobbsbeskrivning

[2]　　　　　CC Systems, <2005-10-03>
　　　　　　　http://www.cc-systems.com/

[3]　　　　　Geographic Information Systems as an Integrating Technology <2005-10-04>
　　　　　　　http://www.colorado.edu/geography/gcraft/notes/intro/intro.html

[4]　　　　　George B. Korte P.E, The GIS BOOK 4 th edition updated and expanded (1997)
　　　　　　　ISBN 1-156690-127-8

[5]　　　　　Mobile GIS ref1<2005-10-05>
　　　　　　　http://www.esri.com/software/arcgis/about/mobile.html

[6]　　　　　Elektronik I Norden nr 17/2004, <2005-10-17>
　　　　　　　http://www.edtnscandinavia.com/tek/showArticle.jhtml?articleID=51201405

[7]　　　　　Microsoft Windows CE 5.0 overview, <2005-10-17>
　　　　　　　http://msdn.microsoft.com/library/default.asp?url=/library/en-
　　　　　　　us/wceappdev5/html/wce50oriDevelopingApplication.asp.

[8]　　　　　Microsoft Windows Mobile, <2005-10-19>
　　　　　　　http://www.microsoft.com/windowsmobile/about/default.mspx

[9]　　　　　Windows CE - Wikipedia, the free encyclopedia <2006-03-13>
　　　　　　　http://en.wikipedia.org/wiki/Windows_CE#Features

[10]　　　　Tommy Niittula, Windows Mobile-programmering,
　　　　　　　ISBN 91-44-03618-3

[11]　　　　Application Development Landscape for Windows CE.NET, <2005-10-11>
　　　　　　　http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dncenet/html/apdvlscape.asp

[12]　　　　Choosing a Windows Embedded API:s vs the .NET Compact Framework, <2005-10-13>
　　　　　　　http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dncenet/html/choose_api.asp

[13]　　　　Microsoft: Classes Exclusive to the .NET Compact Framework (Smart Device Projects), <2005-10-17>
　　　　　　　http://msdn.microsoft.com/library/default.asp?url=/library/en-
　　　　　　　us/dv_evtuv/html/etconclassesexclusivetonetcompactframework.asp

[14]　　　　Andrew Troelsen, Pro C# 2005 and the .NET 2.0 platform,
　　　　　　　ISBN –13 978-1-59059-419-3

[15]　　　　 C# kurshemsida Malmö högskola, <2006-01-11>

[16]　　　　NOAA's National Ocean Service: Geodesy, <2006-03-21>
　　　　　　　http://www.nos.noaa.gov/education/kits/geodesy/media/supp_geo09b4.html

[17]　　　　Pseudorange - Wikipedia, <2006-03-15>
　　　　　　　http://en.wikipedia.org/wiki/Pseudorange

[18]　　　　GPS: the key off it all, <2005-10-12>
　　　　　　　http://www.eurofix.tudelft.nl/gpsintro.htm

[19]   **Master thesis, <2005-10-20>**
http://epubl.ltu.se/1402-1617/2002/141/LTU-EX-02141-SE.pdf

[20]   **Garmins GPS guide for beginners, <2005-10-11>**

[21]   **Geodesi, GPS och detaljmätning | Lantmäteriet**
http://www.lantmateriet.se/templates/LMV_Entrance.aspx?id=68

[22]   **GLONASS – Wikipedia, <2006-02-23>**
http://sv.wikipedia.org/wiki/GLONASS

[23]   **General GLONASS, <2006-02-23>**
http://www.glonass-center.ru/frame_e.html

[24]   **European Space Agency (ESA), <2006-02-21>**
 http://www.esa.int/SPECIALS/Galileo_Launch/index.html

[25]   **Galileo – Wikipedia, <2006-02-28>**
http://sv.wikipedia.org/wiki/Galileo

[26]   **The National Marine Electronics Association, <2006-03-16>**
www.nmea.org

[27]   **Common GPS NMEA sentences, <2006-03-16>**
http://www.commlinx.com.au/NMEA_sentences.htm

[28]   **Nationall_Marine_Electronics - Wikipedia, <2006-03-16>**
http://de.wikipedia.org/wiki/Nationall_Marine_Electronics

[29]   **Geometric Aspects of Mapping, <2006-03-22>**
http://kartoweb.itc.nl/geometrics/index.asp

[30]   **Lars Eklundh, Geografisk informationsbehandling, Metoder och tillämpningar, Tredje reviderade upplagan 2003, ISBN 91-540-5904-6**

[31]   **Jordens koordinatsystem – Wikipedia, <2006-03-02>**
http://sv.wikipedia.org/wiki/Jordens_koordinatsystem

[32]   **IMAGI - Thema des Monats, <2006-03-21>**
http://www.imagi.de/de/thema/c_thema_darstellung_erde.html

[33]   **Coordiantes_datums_projections, <2006-03-06>**
http://www.fws.gov/southeast/gis/training_2k5/coordinates_datums_projections_APR_04.ppt

[34]   **Geodetic services, Overview, <2006-03-10>**
http://www.osi.ie/gps/overview/index.asp

[35]   **Ola Hall, Göran Alm, Stefan Ene, Ulf Jansson, Introduktion till kartografi och geografisk information (2003), ISBN 91-44-04050-4**

[36]   **Ian Heywood, Sarah Cornelius, Steve Carver, An introduction to Geographical Information Systems, 1998, ISBN 0582 08940 9**

[37]   **GIS monitor: Newsletter Archive - 2005 September 1, <2005-10-26>**
http://www.gismonitor.com/news/newsletter/archive/090105.php

[38]   **ADV: Carmenta announces SpatialAce® Light Edition (LE), <2005-10-26>**
http://www.1-script.com/forums/ADV-Carmenta-announces-SpatialAce-Light-Edition-(LE)-article1522-21.htm

[39]   **SpatialAce, <2005-10-26>**
http://www.spatialace.com

[40]   **Anders Forssell på SpatialAce**
Phone: +46 31 775 5700

[41]   **Press Releases - Directions Magazine, <2005-11-02>**
http://www.directionsmag.com/press.releases/index.php?duty=Show&id=11378&trv=1&PRSID=cfa84a291b4adc5
9fec818c75290dc85

[42]   **GISCafe, <2005-11-02>**
http://www10.giscafe.com/link/display_detail.php?link_id=679

[43]   **MapInfo MapX Mobile, <2005-11-02>**
http://www.mapinfo.com

[44]   **ArcPad, <2005-11-08>**
http://www.itc.virginia.edu/research/esri/arcpad.html

[45]   **ESRI, <2005-11-08>**
http://www.esri.com/

[46]   **What's New at ArcPad 6 from ESRI, White Paper, <2005-11-08>**
http://whitepapers.zdnet.co.uk/0,39025945,60047610p-39000701q,00.htm

[47]   **Franson - Quality software at affordable prices, <2005-12-01>**
http://www.franson.com

[48]   **Graticule - Deriving Information from Location Solutions, <2005-11-28>**
http://www.graticule.com

[49]   **GeoFrameworks, <2005-11-14>**
http://www.geoframeworks.com

[50]   **Mobile Application Development eHow-tos and Tutorials <2006-03-13>**
http://www.msdn.net/mobility/windowsmobile/howto/ehowtos/default.aspx#ehowto

[51]   **Geobusiness <2006-04-12>**
http://www.esbic.ie/geobusiness/Mobile_GIS/Main_Components.htm#gis

	Student	Doc Nr		Säk klass
	Markus Pettersson			
	Supervisors on CC-System	Date	Rev	Filename
	Johan Persson and Jörgen Martinsson	2006-05-15		GIS platforms for WinCE

CC SYSTEMS

APPENDIX A

Lab environment
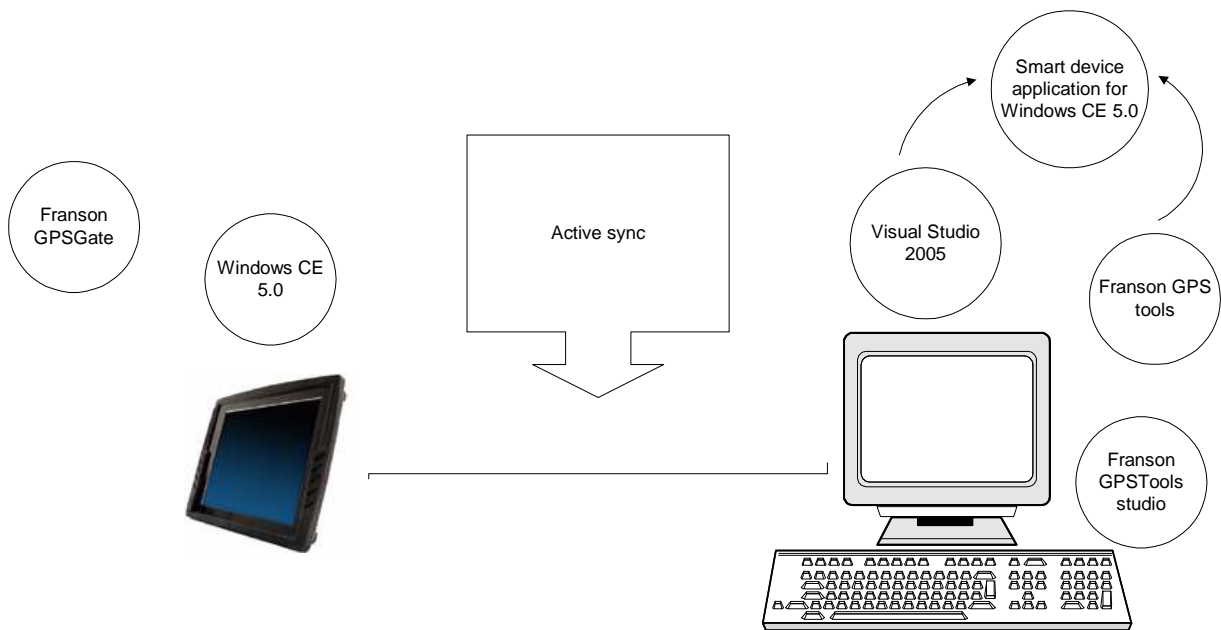
Figure 41. The lab environment

Figure 41 depicts the lab environment. The development has been done in Visual Studio 2005, where a smart device application for C# and Windows CE has been used. Functionality from Franson GpsTool has been integrated in the solution. Franson GpsTools Studio is a software used to define the raster maps to the format used in GpsTools (MapLib). For more information of how this is done see Appendix C.

The communication between the desktop computer and the on board computer is handled by Microsoft Active sync.

The onboard computer CCP XS run windows CE 5.0 and for the time of development a software, Fransom GpsGate, have simulated the GPS.

Microsoft ActiveSync

Since there is no emulator for Windows CE 5.0 included in Visual studio, we debugged the application direct on the on-board computer. Microsoft ActiveSync 4.0 is used to create a connection between desktop and the on board computer. ActiveSync is used to transfer the data and to synchronize the desktop computer and PDA:s.

To establish a connection, start Microsoft ActiveSync on the desktop computer.

The first time established a connection, a set up partnership between the device and the desktop must be created (Figure 42).

Figure 42. Set up a partnership

Choose what should be synchronized (Figure 43), press next.

.
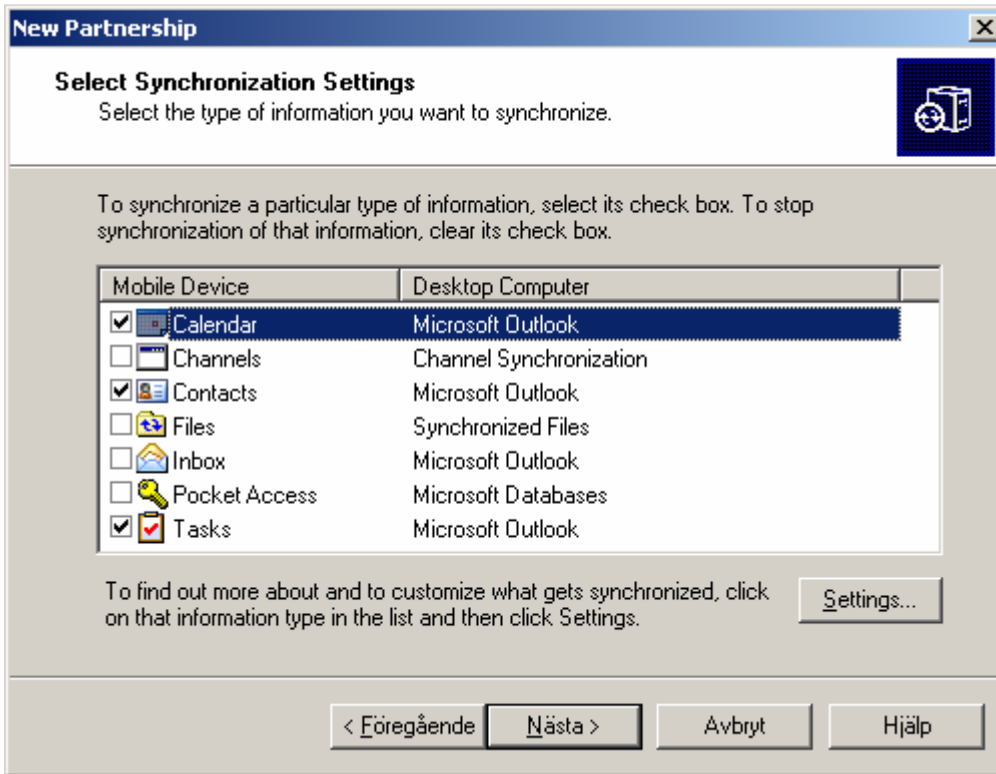
Figure 43. Synchronization settings

From File menu choose get connected (Figure 44)
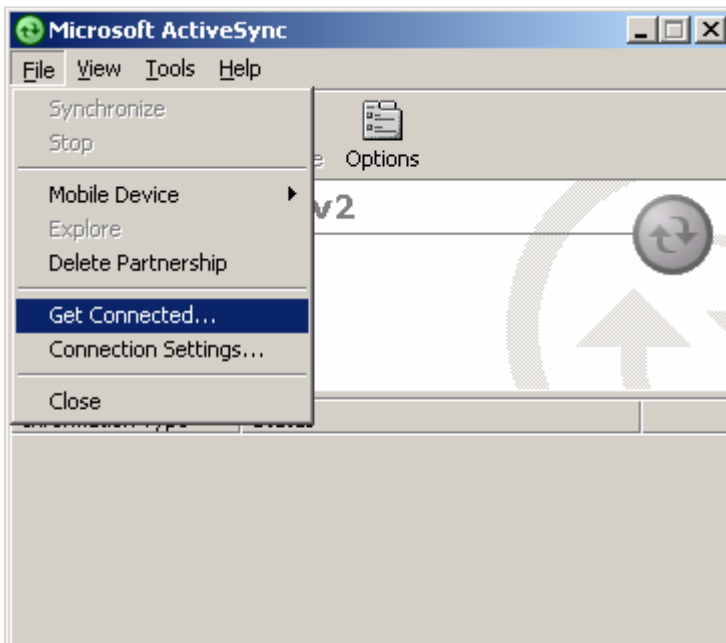
Figure 44. Get connected

	Student	Doc Nr		Säk klass
	Markus Pettersson			
	Supervisors on CC-System	Date	Rev	Filename
	Johan Persson and Jörgen Martinsson	2006-05-15		GIS platforms for WinCE

CC SYSTEMS

When the dialog in Figure 45 shows up, click next. At the same time one click next, execute a program on the on-board computer (see Figure 46).

Figure 45. Get connected dialog

Run the program repllog on the on-board computer (PDA).
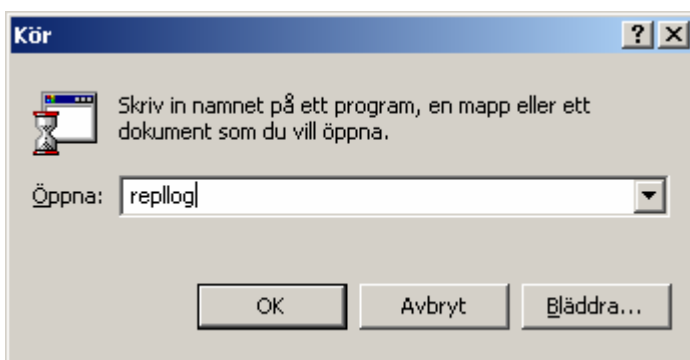
Figure 46. Run repllog

The connection should now be established (Figure 47).
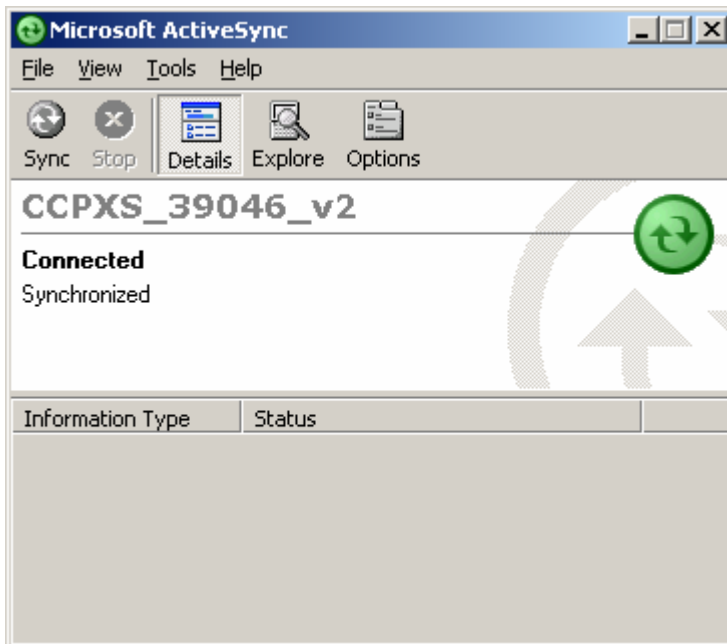
Figure 47. Connection established

Alternatives to Microsoft active sync

For those how does not want to involve Active sync which demands a serial connection, there are other alternatives, one is described on:
http://msdn2.microsoft.com/en-us/library/ms228708.aspx
By using this method it is possible to debug without any serial connection.

Franson GpsGate

GpsGate is software from Franson and has been used to simulate a GPS, i.e. transmit NMEA 0183 sentences on the COM port. This product has been very helpful during the development since it is hard to receive GPS signals from the real GPS indoors. This software has also been used during demonstrations to simulate a route.

APPENDIX B

User manual for the demonstrator XS Navi

Connection of the GPS

Info about current speed over ground

Record route

Info about current position

Action to perform

Zoom in

New shape object

Save shape

Handle layer

Rotate map view

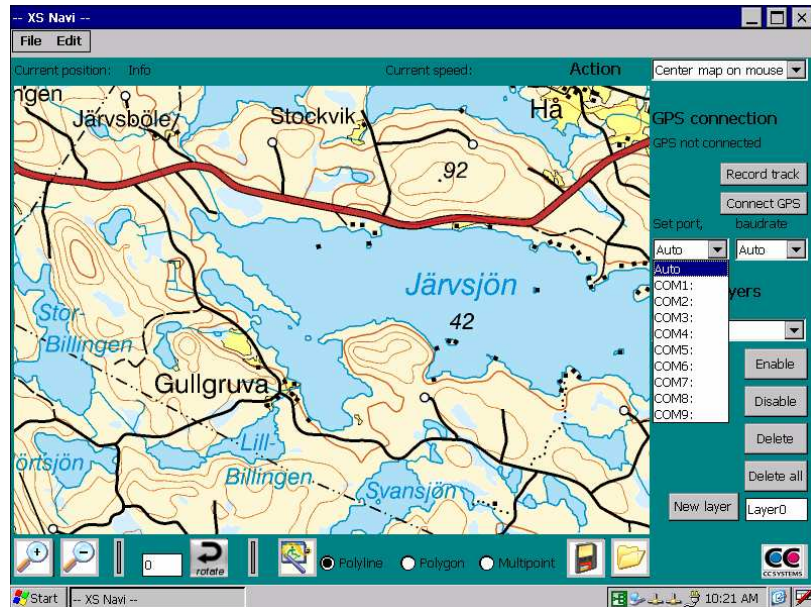Type of shape

Open shapefile

Zoom out

Connect the GPS

- Choose COM2 in the drop-down labelled Set port.

- Set the baud rate to 4800 in the drop-down labelled baudrate.

- Press the connect GPS button.

Or

- From file menu choose Connect the GPS.

- No settings needed, default settings for CCP XS used.

Or

- Choose Auto in one or both of the drop-downs

However, the last choice can led to disconnection of other devices connected to another COM port. This method should therefore only be used if no additional devices are connected to another COM port.

When a connection has been established and a fix is received the map view is centred on the current position. The position is marked with a black rectangle.
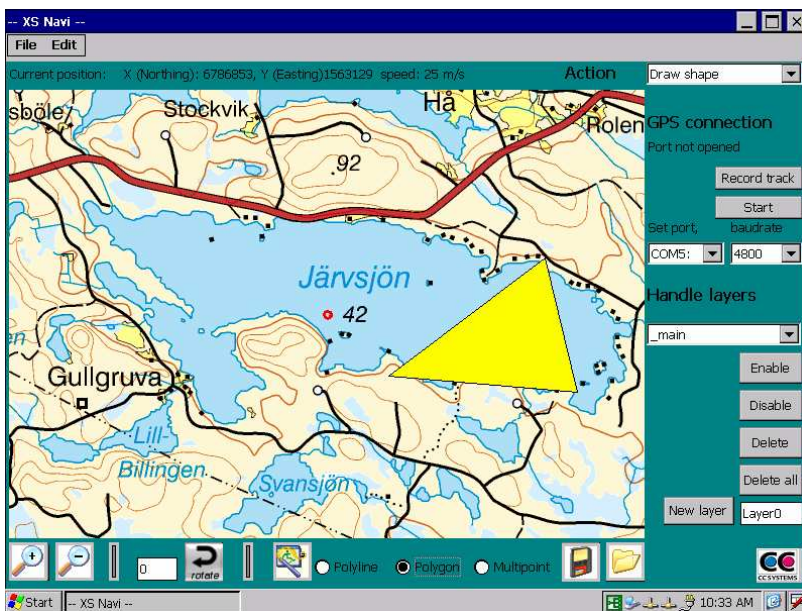
If something is wrong with the connection or if no fix is made, the user is informed.
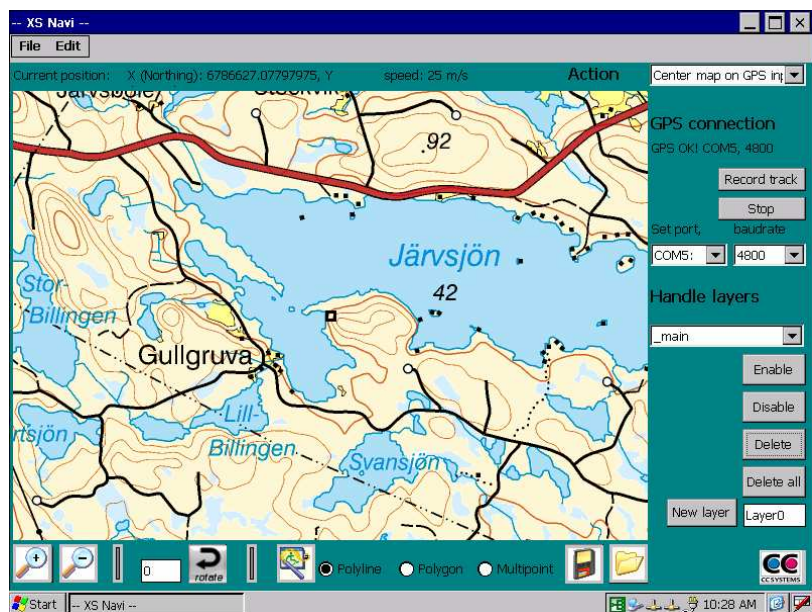
Draw shape
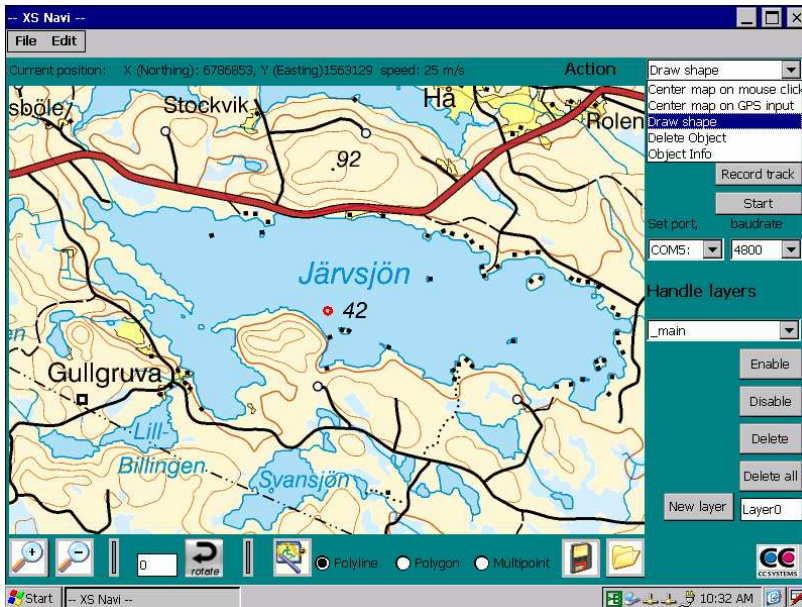
- From the drop-down labelled Action, choose Draw shape

- Click on the new shape icon
- Choose shape type, polyline, polygon or multipoint
- Draw

Save Shape

The current drawn shape is saved to an ESRI shapefile. On click a Save file dialog is displayed.

Open Shapefile

On click an open file dialog is displayed. The chosen shapefile is displayed in the current layer in the map view.

Record track

On click the current route is recorded. The track is saved to a default shapefile "program files\xs_navi\Shapefiles\track.shp" when the user clicks the button the second time.
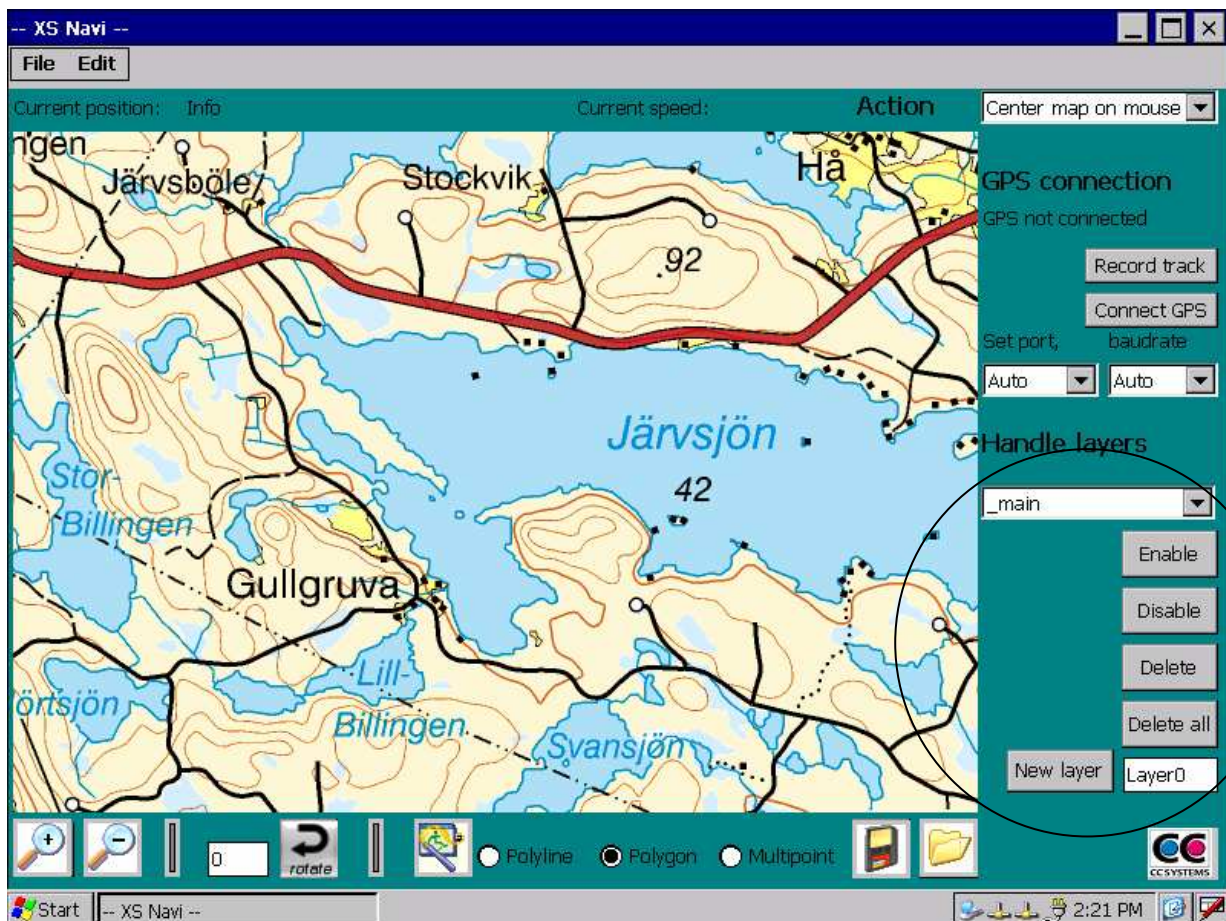
Zoom

The zoom functionality in the demonstrator only affects the size of the pixels. The resolution is the same. I.e. when the zoom in button is clicked a smaller area is displayed but the reslution is the same.

Rotate

The map view is rotated with the number of degrees given in the textbox.

Student	Doc Nr	Säk klass
Markus Pettersson		

Supervisors on CC-System	Date	Rev	Filename
Johan Persson and Jörgen Martinsson	2006-05-15		GIS platforms for WinCE

Handle layer

- New layers can be created on top of the background map. The meaning of background map is in this case the map that has been opened at first. By clicking the New Layer button a layer, with the name written in the textbox, is created.
- A layer is active when the user selects it in the drop down menu.
- The delete all button deletes all layers except the background map (MapLib).
- The delete button deletes the active layer.
- The enable and disable buttons, enable or disable the active layer.

	Student Markus Pettersson	Doc Nr		Säk klass
	Supervisors on CC-System Johan Persson and Jörgen Martinsson	Date 2006-05-15	Rev	Filename GIS platforms for WinCE

CC SYSTEMS

APPENDIX C

Define raster map in GpsTools Studio

GpsTools Studio is used to create and edit MapLibs. A MapLib contains a raster map and icons to be used in your application.

After installing GpsTools SDK 2.0 (or later) you can find GpsTools Studio under
`Start->Programs->GpsTools SDK->GpsTools Studio`

Create a new MapLib
You need to have a raster map in GIF or BMP format.

To create a map from a bitmap:
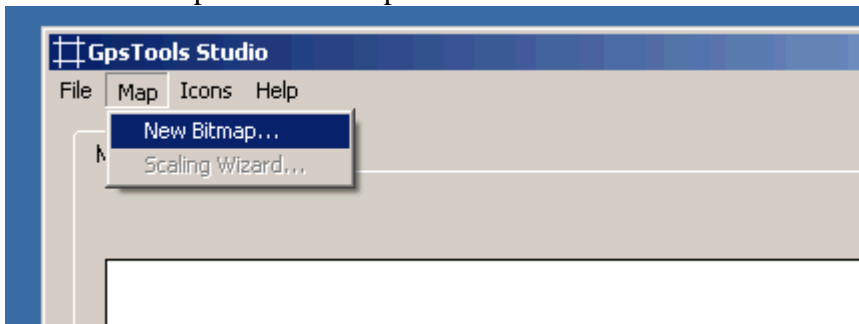
The raster map must be **conformal.**(*) This means lines from south to north must be orthogonal to lines that goes from west to east. The map can be rotated. The scale can be different (or same) in north/south direction than west/east direction.

After the bitmap is loaded a wizard will start to scale the raster map. What you need is three positions on the map with know geographic coordinates. You also need to know which coordinate system the map is using. If you are unsure use UTM, and use the method presented under "Verify scaling" below to see if you made the right choice.
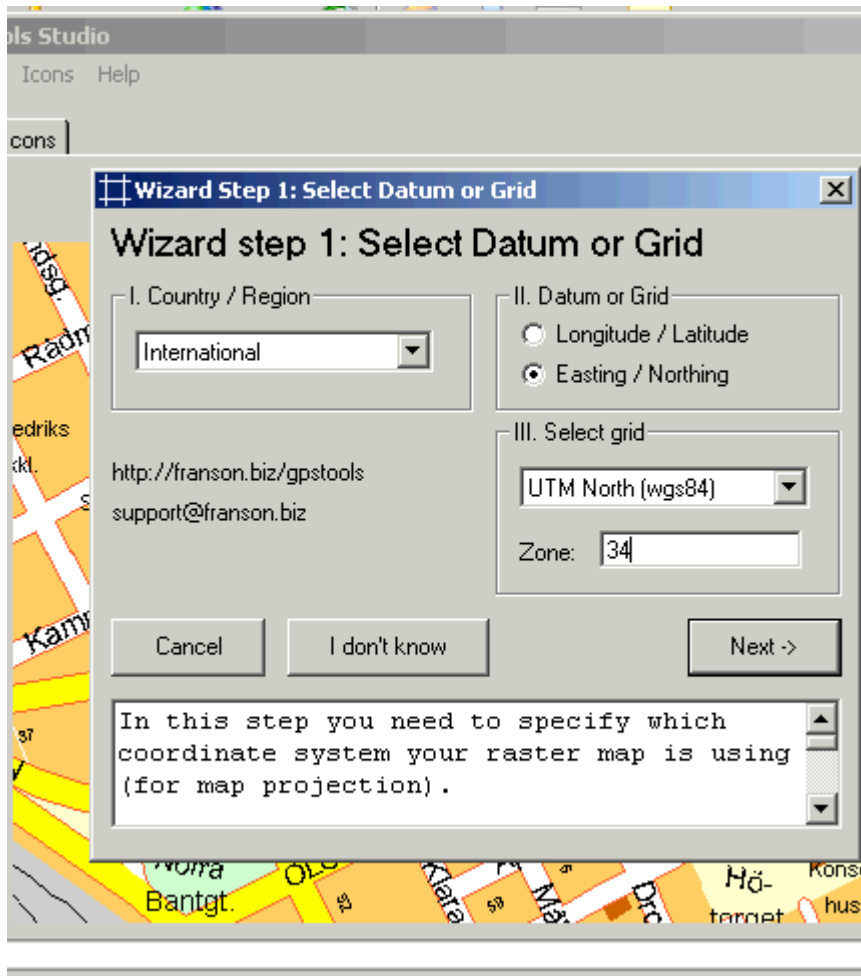
Wizard step 1 - Select Datum or Grid.
In this step you need to specify which coordinate system your raster map is using.
I. Select a Country / Region.
II. Select if the coordinates are specified in lat/long or easting/northing
III. Select the grid or datum.
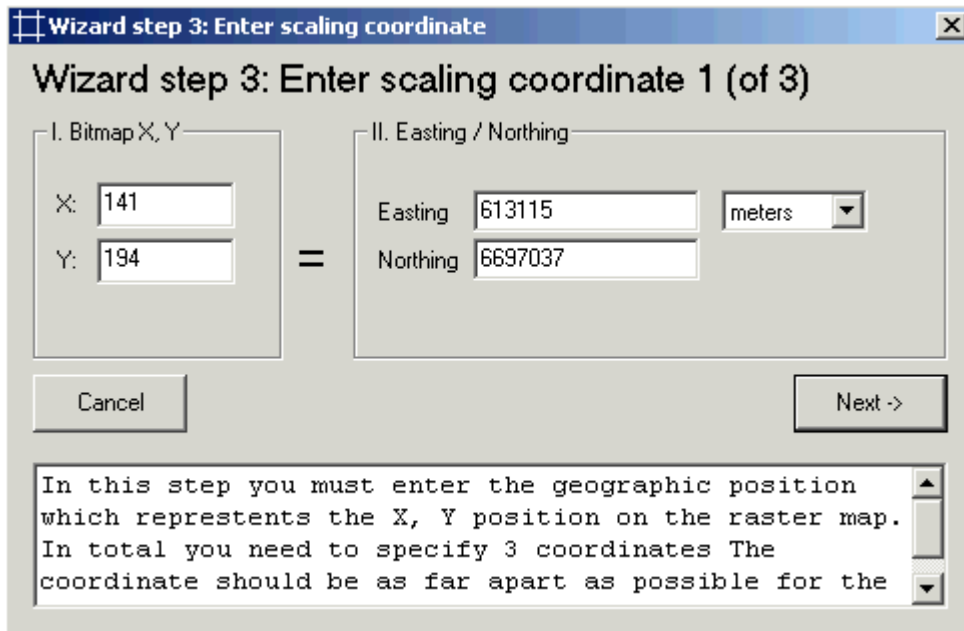The coordinate system chosen is that in which the map is conformal.

Enter coordinates

After selecting coordinate system, click somewhere on the map and this dialog shows up:

You must enter the geographic position, which represents the X, Y position on the raster map. In total you need to specify 3 coordinates The coordinate should be as far apart as possible for the best results. Try to place them in opposite corners.
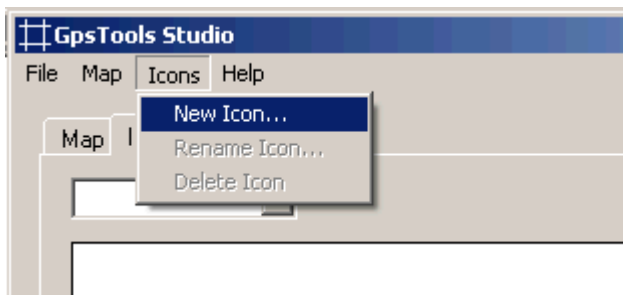
Verify scaling
To verify that everything is ok, click on some other known position on the map and verify that the geographic coordinate show in the bottom line of the window is correct.

Create Icons
You can create icons (bitmaps), which you later can be drawn on the map in your application.

To create a new icon select:

The Icon's bitmap must be a BMP of GIF. If the GIF has a transparent color, it will be transparent in the MapLib as well. After loading the bitmap, you need to name the Icon. This name will be used to refer the Icon in your application. See `Map.NewIcon`
Any number of Icons can be created. Use the menu to rename, delete and create new Icons.

Save MapLib
After creating a Map and Icons, you need to save the MapLib:
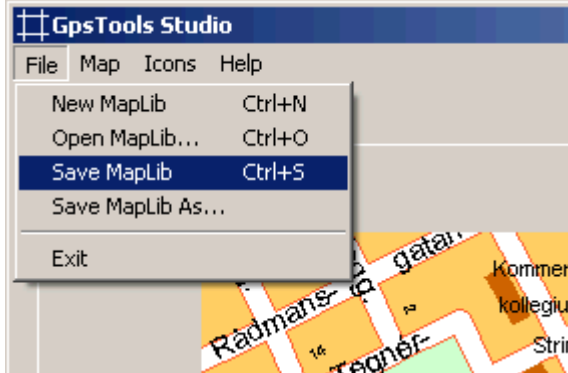
Now the MapLib is ready to be used by your application. See `Map.Open`.

(*)Conformality expressed in a more formal way:
When the scale of a map at any point on the map is the same in any direction, the projection is conformal. Meridians (lines of longitude) and parallels (lines of latitude) intersect at right angles. Shape is preserved locally on conformal maps.

APPENDIX D

Shortening

CC Systems	Cross Country Systems (company name)
CCP XL	Cross Country Pilot XL, PC-based onboard computer
CCP XS	Cross Country Pilot XS, Onboard computer running Win ce.NET
API	Application Programming Interface
GUI	Graphical User Interface
GIS	Geographic Information System
GPS	Global Positioning System
ESRI	Environmental System Research Institute
PDA	Personal Digital Assistent
Windows CE.NET	Operating system from Microsoft for mobile or smart devices
NMEA	National Marine Electronics Association
NMEA 0183	Protocol from NMEA
.NET CF	.NET Compact Framework