

Reports in Geodesy and Geographical Information Systems

Distribution of GPS-data via Internet

Martin Peterzon

Thesis work

Gävle 2004

L A N T M Ä T E R I E T



Abstract

The progress of easy mobile Internet access has triggered research activities around new methods of distributing GPS-data in high accuracy positioning systems. National Land Survey of Sweden, Lantmäteriet, operates a network of permanent reference stations, SWEPOS™, where Internet-distribution of GPS-data to users is a subject of interest. This paper shows that such a distribution is possible and that there are many benefits in the technique. The main focus is on the real-time protocol Ntrip, developed for GPS-data – its components and data format.

A test system for Ntrip and an application that monitors the quality of DGPS-data was implemented and tested at Lantmäteriet, showing promising results with low latencies, good reliability and simple installation procedures. Ntrip is flexible; hence two approaches of integration with the present SWEPOS system and three solutions of the communication between different Ntrip system are proposed. Cost aspects are also investigated. With good Internet provider contracts and the new, compact data format in use, Internet distribution will become more cost-efficient than the systems used today.

Preface

This report describes the thesis work that finishes my Master of Science in Information Technology Engineering at Uppsala University. The work has been carried out at the Geodetic Research Department at the Nation Land Survey of Sweden, Lantmäteriet, in Gävle.

I would like to thank my supervisors Gunnar Hedling and Bo Jonsson, along with everybody else at the Geodetic Research Department, for being friendly and helpful. I would also like to thank Georg Weber, BKG, for active participation in the thesis; Göran Arrhen, Trimble, for software support; and my examiner Ivan Christoff at the Department of Information Technology, Uppsala University, for valuable feedback.

Uppsala, January 2004

Martin Peterzon

Table of Contents

1. Introduction.....	1
1.1. Background	1
1.2. Problems	1
1.3. Disposition	1
2. GPS and Networks.....	2
2.1. GPS and Corrections	2
2.2. Network RTK.....	2
2.3. SWEPOS	3
2.4. Data Flow	4
2.5. TCP/IP	4
2.6. Distribution of Corrections.....	5
3. Networked Transport of RTCM via Internet Protocol.....	5
3.1. System Overview	6
3.2. NtripCaster.....	7
3.3. NtripSource	8
3.4. NtripServer.....	8
3.5. NtripClient.....	8
3.6. Messages.....	9
4. Test System.....	9
4.1. Local RTCM Source	10
4.2. Stationary BKG Client.....	10
4.3. Mobile BKG Client	11
4.4. Trimble ACU Client.....	11
4.5. Latency Measurements.....	11
4.6. Conclusions	14
5. Broadcast corrections	14
5.1. Broadcast versus Virtual Reference Station.....	14
5.2. RTCM SAPOS.....	16
5.3. Broadcast in GPSNet.....	16
6. Integration in the Present System	16
6.1. Local NtripCaster.....	16
6.2. RTCM Sources	17
6.3. New Rovers.....	18
7. Cost and Data Amount	19
7.1. Frequency.....	19
7.2. RTCM Size.....	19
7.3. Cost of GSM and GPRS	20
7.4. Future Considerations.....	21
7.5. Conclusions	22

8. Communication Between NtripCasters	22
8.1. Modified NtripClient	22
8.2. Combined NtripServer and NtripClient	23
8.3. Modified NtripCaster	24
9. Monitoring the Quality of Corrections.....	25
9.1. Correction Quality Parameters	25
9.2. Components and Communication	26
9.3. Implementation	27
Local Test.....	27
10. Conclusions.....	28
11. References	30
Appendix A – Configuration of the built-in NtripClient in Trimble SC.....	32
Appendix B – Size of RTCM 2.3 data packets	33
Appendix C – Setting up Logger and Monitor programs	35

1. Introduction

1.1. Background

SWEPOS™ is a network of permanent reference stations operated by National Land Survey of Sweden, Lantmäteriet, providing high accuracy positioning and navigation for scientific and practical use. Today distribution of GPS-data to users is handled with FM radio, geostationary satellites, GSM and FTP, but there is an increasing interest in using Internet as communication channel. With good mobile coverage and cost-effective methods it is a distribution technique for the future.

Networked Transport of RTCM via Internet Protocol (Ntrip) is an application level protocol for distribution of GPS-data. Test systems in Germany indicate that the protocol could also be efficiently implemented in SWEPOS. A study of the possibilities to use Internet distribution with Ntrip at Lantmäteriet is of great interest.

1.2. Problems

This paper will present a possible establishment of Internet distribution and Ntrip within SWEPOS. Several aspects are considered. Issues like (1) hardware and software requirements, (2) distribution techniques, (3) integration with present system, (4) flexibility, and (5) costs will be discussed. With the test system that was set-up, it is also possible to answer questions concerning (6) packet latencies and (7) performance of the protocol. An implementation of an application that monitors data quality in the SWEPOS network RTK system shows (8) a realization of a computer network system handling GPS-data.

1.3. Disposition

Section 2 gives an introduction to GPS, Network RTK and existing positioning systems. It also describes some basics in computer communication. In section 3 the Ntrip protocol is introduced, followed by a description, in section 4, of a test system, with details about different configurations and latency measurements. Section 5 deepens the discussion of broadcast methods of corrections, and section 6 suggests different approaches that can be used in the integration of the present system. Packet sizes of different correction messages are illustrated in section 7, followed by a comparison of costs of Internet distribution and methods used today. Section 8 suggests different approaches for communication between NtripCasters – the centers of the distribution. An implementation of an application that monitors the quality of Network RTK corrections is described in section 9. This is an example of a service that can use computer networks for GPS-data distribution. Finally, in section 10, the paper is summarized and conclusions are presented.

2. GPS and Networks

To improve the accuracy in GPS (Global Positioning Systems) positioning several different methods exist. They are all based on information sent between the measurement equipment and other devices, such as permanent reference stations. Below is a short description of GPS and some of these techniques. Since this paper is about distribution of data via computer networks, a brief description of basic computer communication is also included.

2.1. GPS and Corrections

GPS is an American positioning and navigation system consisting of 29 satellites (January 2004). It was developed mainly for military purposes. By knowing the distance to at least four satellites it is possible to calculate a position and a time offset almost anywhere on earth. If two GPS-receivers have visual sight of the same satellites, a relative positioning can be performed, where the differences between the stations are measured. This reduces most error sources. In relative positioning, usually one station is at a fixed location (the base) and the other is moved between points to be determined (the rover).

The two most common methods for real-time relative positioning are called DGPS (Differential GPS) and RTK (Real-Time Kinematic). DGPS uses signal travel time between satellite and receiver to calculate the distance, while RTK also uses a carrier phase and the distance is measured in number of full phase cycles plus a decimal part of one period. The accuracy is 0.5 - 2 meter for DGPS and 1 - 5 centimeter for RTK.

There are a number of error sources in a GPS measurement. Errors in satellite orbit and clock are eliminated in relative positioning, while ionospheric, tropospheric, and receiver clock errors still remain, but are reduced.

For survey positioning, and many other types of positioning, the accuracy that one gets from relative methods is necessary. A network of fixed GPS-receivers, which can be used as bases, is therefore useable. Such a network exists in Sweden: SWEPOS. The bases are called permanent reference stations and users have the possibility to utilize them as their base in relative positioning. Messages sent from a reference station to a rover are called corrections. The most common data format for corrections is RTCM – a recommended standard developed by the Radio Technical Commission for Maritime Services [R01].

2.2. Network RTK

There has been an increasing interest in the concept of Network RTK during the last years. Measurements can be done with improved integrity, continuity and accuracy in a more sparse network of stations. It also leads to productivity improvements – since only one receiver is necessary and no temporary base station have to be established – and the cost of establishment and maintenance is reduced. For a working Network RTK configuration the stations have to be connected to a control center with a data link.

The basic idea is to use information from all reference stations in the network, instead of the closest only (as in DGPS and RTK). One reference station works as a central unit, collecting data from all the stations in the network. By using information from the whole area, it is possible to use models that make better estimates of the errors. The corrections

can then be sent from the central reference station – here called the control center – to rovers in the network.

Basically there are two different methods for distributing the corrections in Network RTK: VRS (Virtual Reference Station) and Broadcast. Both of these are described below.

2.2.1. Virtual Reference Station

VRS requires bi-directional communication between the rover and the control center. It needs no new message types since everything sent works like standard correction messages in DGPS or RTK.

A virtual reference station is a simulation of a reference station. At any position in the network's coverage area, the control center can approximate the correction data that a reference station would send if it were located at that position. The control center uses information from all other stations to compute these corrections.

To start with, the rover sends its approximate position, from a standalone GPS measurement (1), in a NMEA GGA message to the control center. The control center replies with RTCM correction (2) for this position, which is used by the rover to compute a DGPS solution. The new, improved position – that is sent back to the control center (3) – is then used as the position for the virtual reference station. Again, the control center sends out correction data (4). This time the quality is as if the reference station was right beside the rover, and the DGPS calculation will give even better values.

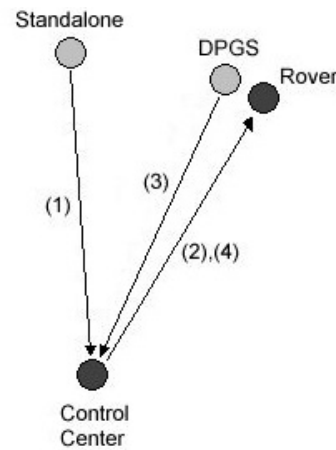


Fig 1. VRS communication

2.2.2. Broadcast

Broadcasting the corrections to the rovers requires new messages to be defined; the ones used in DGPS or RTK are not enough. It is a one-way communication process with the control center as a sender and the rover as a receiver.

In broadcast mode the control center estimates correction parameters valid for the area between reference stations and then sends them out. A rover will use parameters that correspond to its closest environment. One usually works with different models and parameters of the ionosphere, where the broadcast message includes variable values for this model.

2.3. SWEPOS

SWEPOS is a network of permanent reference stations [wSWEP] run by Lantmäteriet, Nation Land Survey of Sweden. It has been in operational mode since 1998, providing navigation and positioning service with 1-meter level accuracy in real-time measurements using DGPS. If one uses post-processing data from several reference stations, centimeter level accuracy is possible. Today (January 2004) the network consists of 57 stations.

During the last years there have been several pre-study projects on Network RTK in Sweden [JHW03]. These studies have shown very promising results, both concerning the performance and user interest; and in January 2004 the service SWEPOS Network RTK became available to the public. At present the network covers southern Sweden, the area around Gothenburg and the Mälardalen region.

2.4. Data Flow

All stations within the SWEPOS network are connected to the control center, geographically located in Gävle. Raw observation data and RTCM messages (type 1, 2 and 3) are sent with a frequency of 1 Hz to the control center via leased lines using the TCP/IP protocol (see section 2.5).

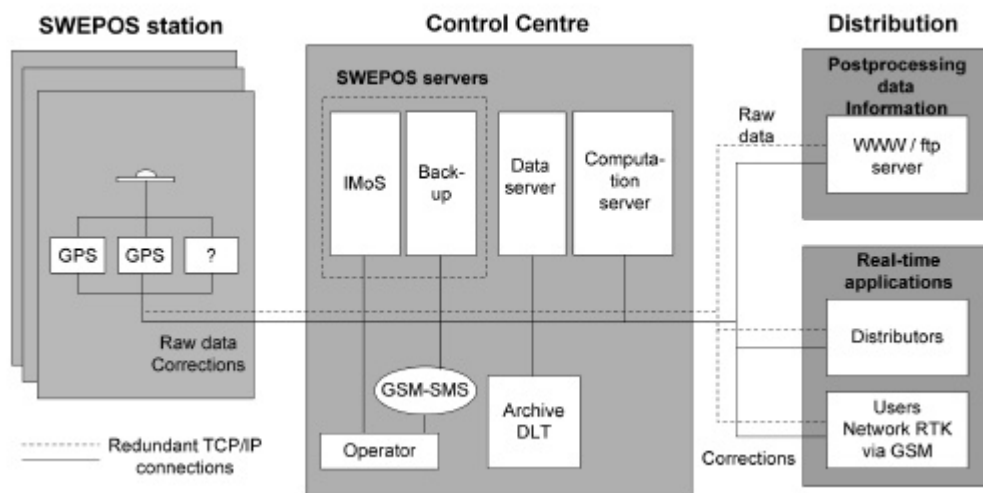


Fig 2. Design of SWEPOS

Users can either fetch raw data in Rinex format from the control center via FTP-servers or receive real-time RTCM corrections via a distribution server. Network RTK data is also distributed through this server via GSM.

The IMoS software [mIM] is used at the control center. It receives data, routes data, verifies the validity, creates files for post-processing and distributes corrections to remote users. It also contains an interface for monitoring the current status of the system, for managing history files and to perform necessary maintenance.

In the Network RTK projects in Sweden, Trimble GPSNet [mGN][wTRIM] has been used as Network RTK and distribution software. Corrections are distributed to users via GSM and the Network RTK mode is VRS.

2.5. TCP/IP

Data communication is traditionally handled in packets. A packet is a chunk of data that is sent from the sender to the receiver. In a majority of computer networks TCP (Transmission Control Protocol) [P81b] and IP (Internet Protocol) [P81a] are the techniques used for data transportation, and as in all modern networks the layer principle

is the fundamental concept, where each layer presents a predefined interface to the layer above it. This makes modular design possible and new interfaces can be easily introduced. TCP/IP is defined by four layers (see figure 3). The link layer controls the interface between the operating system of the computer and the computer network. IP handles packet routing (network layer) and TCP handles packet flow (transport layer).

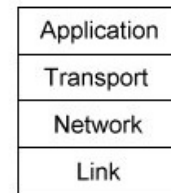


Fig 3. TCP/IP layers

The top level is the application level. This is where the different services for users are located – for example SMTP for mail and HTTP for web sites. This is also where the Ntrip protocol – a solution for transporting GPS corrections via Internet (described in section 3) – is defined.

2.6. Distribution of Corrections

Today real-time DGPS data from SWEPOS is available through FM radio (via the RDS channels) and geostationary satellites. The FM radio transmission is available in most of Sweden while the geostationary satellites have an almost worldwide coverage. Because of low elevation angles of the satellites at high latitudes, the usage of these is limited at land based measurements. GSM have been used as communication channel in Network RTK projects. This gives coverage in big parts of Sweden.

A majority of proposals concerning new means of distribution are based on Internet and IP traffic. One example is a system to access EGNOS (European Geostationary Navigation Overlay Service) correction data over mobile IP [CTV03]. Data from geostationary EGNOS satellites are collected at a base station and transmitted, in real time via Internet, to GPS-receivers. This removes the problem of limited view of the satellites at high latitudes.

IGS (International GPS Service) is another project where Internet is used as the communication channel for GPS-data [wIGS]. Data centers distribute data via UDP (User Datagram Protocol), which is a commonly used protocol working at the same layer as TCP. An application called UDP relay [mUR] can be installed at the data centers providing GPS-data to other data centers and end users. Applications for multicasting are also available within the system.

This paper, though, is focused on the distribution technique Ntrip. It shares many characteristics with IGS but use TCP instead of UDP. Ntrip is described below.

3. Networked Transport of RTCM via Internet Protocol

Networked Transport of RTCM via Internet Protocol (Ntrip) is a protocol for streaming GNSS (Global Navigation Satellite System) data over Internet. It has been developed as a project under EUREF – a commission responsible for the European reference system – by BKG (Bundesamt für Kartographie und Geodäsie), and it has been tested in several configurations. Ntrip is still under development.

Figure 4 describes how Ntrip communication works. The user – that is the NtripClient – receives correction data from the control center via Internet, for example by connecting a mobile phone with GPRS (General Packet Radio Service) to the survey equipment. Data is sent in Ntrip format over Internet, which makes it possible to receive corrections almost in real time from available reference stations. The communication between the control center (NtripCaster) and the reference stations (NtripServers) can also be carried out using the Ntrip protocol. Both users and reference stations receive GPS satellite data as usual.

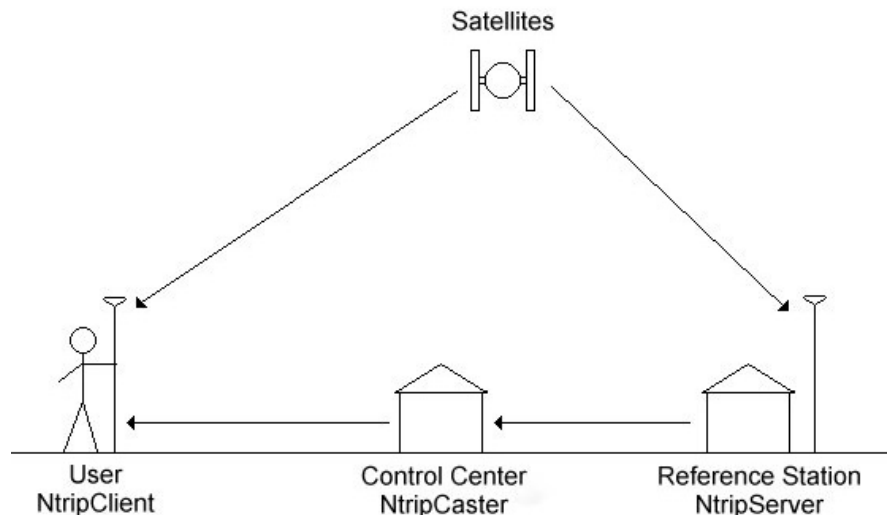


Fig 4. Schematic view of Ntrip communication

The correction provider is responsible for the NtripCaster at the control center. It distributes data to all users and fetch data from all reference stations. This means, from the user's point of view, that the only difference compared to the present Network RTK system operated by Lantmäteriet would be the device type by which corrections are received. Instead of using GSM, the communication with the control center is handled via Ntrip and Internet.

Ntrip messages are sent with the HTTP (Hypertext Transfer Protocol) version 1.1 [F97], which is the application level protocol used for traffic on WWW (World Wide Web). Any GPS-receiver system, that has a connection to Internet/WWW, can therefore be configured to receive for example RTCM data with Ntrip.

Below follows a short description of the protocol. I will assume that the RTCM format is used, although other GNSS data formats are possible. More detailed information can be found in the Ntrip documentation [GW03].

3.1. System Overview

All messages transmitted with Ntrip is either sent or received by the NtripCaster – the center of communication in a Ntrip system (see figure 5). To the NtripCaster one can connect NtripServers that fetch information from one NtripSource – a source that provides RTCM messages – and sends these on to the NtripCaster. NtripClients can then

access a RTCM stream from a NtripSource by requesting it from the NtripCaster. All NtripClients and NtripServers connect to the same IP address, leaving to the NtripCaster to distribute the streaming data.

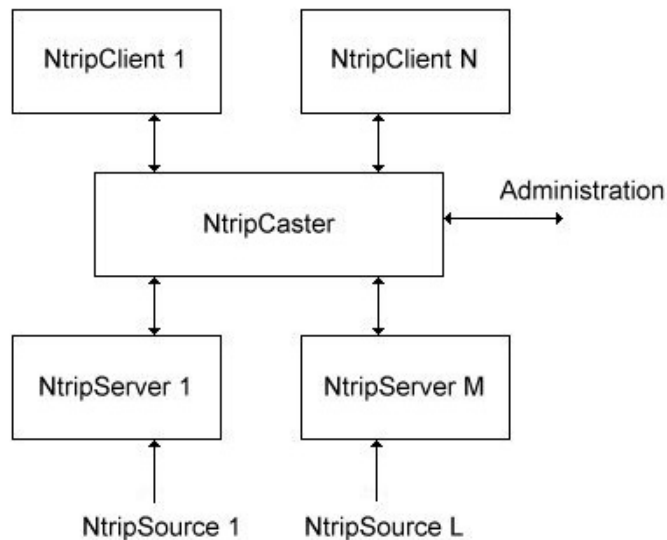


Fig 5. Ntrip system architecture

Administration of the NtripCaster is handled via Telnet.

3.2. NtripCaster

The NtripCaster is the center of communication in a Ntrip system. It collects data from NtripSources and distributes data to NtripClients. Hundreds of devices can be connected at the same time using the common TCP/IP-protocol, and they all have the same access point to the system. By sending either corrections or request for corrections in the right format to an IP number and a reserved port (usually port 80 that traditionally is reserved for HTTP traffic), the NtripCaster can handle the request and do proper actions. Authentication of the users is also possible.

To keep track of all the sources, a source table is maintained in the NtripCaster. This table not only contains information about the sources but also information about networks of sources and other broadcasters. Every source is identified with a unique mountpoint.

A network defined in the source table is a virtual network containing NtripSources. Dividing the sources in different networks not only makes it easier for the user to overview the system, it also provide a method for authentication. An administrator can grant access for a user to all NtripSources in a network. One NtripSource can only belong to one network.

A caster defined in the source table of another caster is purely a reference, telling about its IP address, port number and some other useful information. There can be no direct connection between two NtripCasters, although it would be possible to create a NtripClient that can handle several different NtripCasters at the same time, making it

visible for the users as if the NtripCasters are co-operating (for more details, see section 8.1).

Telnet is used for administration of the NtripCaster. The administrator can, for example, allow and deny clients and sources, monitor different statistics and set authorization.

Technically a NtripCaster is - in most aspects – an HTTP server supporting a subset of HTTP requests and responses adjusted to streaming data. In terms of the standard client/server terminology a NtripCaster works as a server, while both a NtripClient and a NtripServer works as clients. This means that the NtripCaster is always the passive device in the connection, the one that is contacted.

3.3. NtripSource

A NtripSource is a geographically stationary point that provides continuous, streaming RTCM data. This data is then sent to the NtripServer, which is taking care of the transfer to the NtripClient.

A NtripSource is identified by a unique mountpoint in the source table of a NtripCaster. It is by this mountpoint that a NtripClient can get access to the source. The mountpoint is the single thing identifying the source and is the key value when establishing a RTCM stream through the NtripCaster. The source table also specifies the RTCM format and some other characteristics.

3.4. NtripServer

A NtripServer receives RTCM data from a NtripSource and sends it to the NtripCaster. When setting up the NtripServer one needs to know a mountpoint and a password; both are provided by the administrator of the NtripCaster. This information can not be sent through the Ntrip system, but have to be delivered by some other medium – for example e-mail.

Because TCP is able to detect broken connections, a NtripServer can be programmed to deal with such events, for example by trying to reconnect. TCP also makes it possible for data to travel through most parts of Internet without firewalls interrupting the traffic.

An example of a NtripServer implementation in Windows is provided by BKG in Germany [wBKG].

3.5. NtripClient

The NtripClient is the component that is installed in the users GPS receiving device system, for example in a pocket PC. It requests data from a NtripCaster by asking for a stream from a specific mountpoint, given by the source table it has received from the NtripCaster. Depending on the implementation of the NtripClient this stream can be handled in different ways. For example one could transfer it with a serial cable to a GPS-receiver or monitor it on the computer directly using a RTCM interpreting application.

As with the NtripServer, the NtripClient can be programmed to deal with broken connections using the support for this in TCP.

For some GPS techniques, such as VRS, the NtripClient has to be able to send its position to the NtripCaster. This is supported in Ntrip with the possibility to send NMEA GGA strings attached to a HTTP message. It is specified in the source table whether the NtripSource use NMEA messages or not.

Examples of NtripClient implementations for Windows/Windows CE/Linux are provided by BKG [wBKG] (see figure 6). In these clients BKG have chosen not to support NMEA request messages. Trimble supports NMEA in their system.

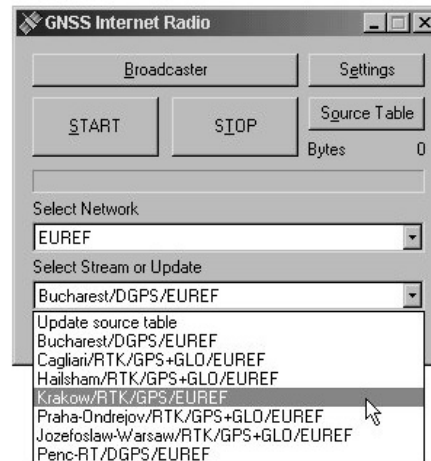


Fig 6. BKG' s NtripClient when the user chooses reference station

3.6. Messages

Messages are sent on TCP/IP connections using the HTTP 1.1 protocol. There are certain messages for the NtripClient-NtripCaster and NtripServer-NtripCaster communication. The communication is sometimes bi-directional.

Server messages deals with the task to establish a connection between the NtripServer and the NtripClient, making it possible to transfer RTCM data. The NtripServer sends a message with mountpoint and password (that has been provided by the administrator earlier). If mountpoint and password match, the NtripCaster establishes the connection, otherwise an error message is sent.

Client messages asks for data from a specific mountpoint. If no mountpoint or an invalid mountpoint is transmitted, the NtripCaster replies with the source table, otherwise a RTCM stream is opened.

Authentication can be used in the client messages. Together with the request a password is attached for that specific source or network. This can for example be used for billing purposes. Two different layers of security can be used: basic authentication scheme and digest authentication scheme. The latter is a more secure authentication algorithm that does not have to send the password in an explicit message.

4. Test System

To test the functionality of the current Ntrip test system, operated by BKG, a GPS-receiver was established locally in Gävle, working as a NtripSource. A NtripServer was connected to the source, constantly sending RTCM messages to the NtripCaster. Different implementations of NtripClients where then tested and connected to this NtripCaster, receiving corrections that could be used for more accurate positioning. Figure 7 describes the test system

To be able to get access to the NtripCaster provided by BKG – except for a few open sources – one has to send in an application. This is also necessary for setting up a NtripSource. Applications can be found at BKG’s web page [wBKG].

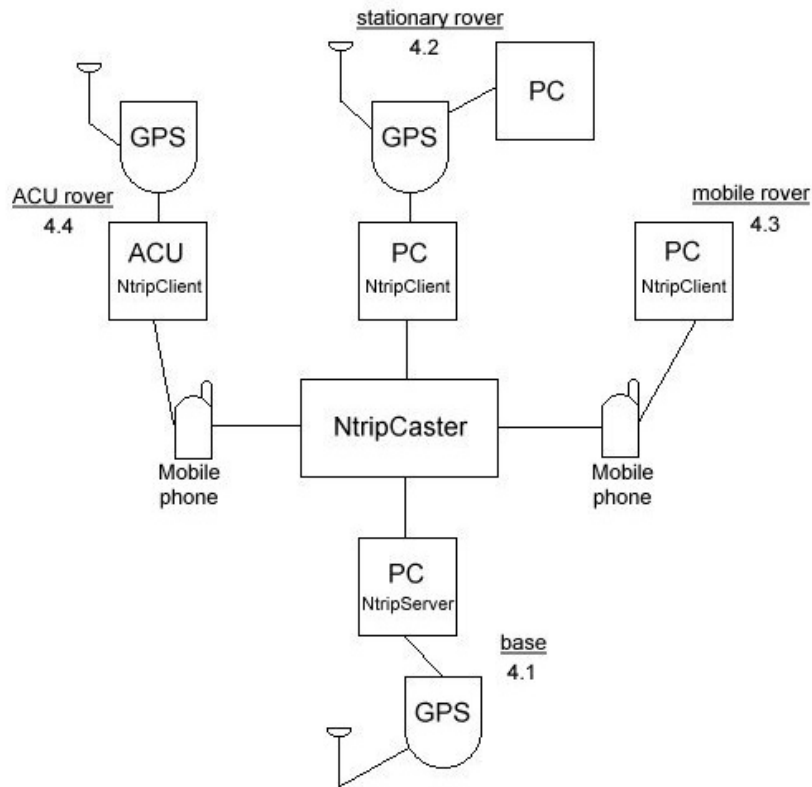


Fig 7. Local Ntrip test system

4.1. Local RTCM Source

A Javad GPS receiver, connected to the local GPS base, was used to calculate the corrections. These corrections were then transmitted over a serial cable as RTCM data to a PC. The PC had a copy of BKG’s NtripServer software, configured with a mountpoint that was known by the NtripCaster. The RTCM messages were then encapsulated in Ntrip packets in the NtripServer and sent to the NtripCaster on demand.

RTCM messages of type 1 and type 31 were used with a frequency of one per second each. They were transmitted over the serial cable with a speed of 9600 bits per second. That is more than enough for the frequencies of RTCM messages used.

4.2. Stationary BKG Client

The first attempt to receive corrections and use them for DGPS calculations was made on a stationary configuration, using the same antenna as both rover and base – a so called

zero baseline. A PC connected to Internet ran the Windows NtripClient provided by BKG, and sent its output on a serial cable to a Javad receiver. The Javad receiver also fetched GPS data, and with the software PCView [mPV] installed on another PC this data could be viewed.

The traffic could then be logged with PCView software, for example by monitoring NMEA GGA messages in the GRIL interface [mGR]. Another way of logging is to catch the RTCM packets when they leave the NtripClient. This was done with the “RTCM Control Program for Ntrip GNSS” for the latency measurements. The program is provided by BKG.

4.3. Mobile BKG Client

An Ericsson T39 mobile phone was connected to Internet with Telia GPRS. The Internet traffic was then transmitted to a laptop via infrared communication, and an installation of BKG’s NtripClient picked up the Ntrip packets and sent them in RTCM format to the RTCM Control Program.

This configuration was used for measurements of latencies of the mobile communication.

4.4. Trimble ACU Client

The Survey Controller (SC), version 10.71, in Trimble’s ACU has a built-in NtripClient. It can download the source table of a NtripCaster and let the user browse among the sources, choosing which station to use. The client also supports authentication of the users. It can handle DGPS, RTK and SAPOS (a broadcast technique for Network RTK, see section 5.2). Earlier versions of the SC only includes support for receiving Internet corrections, but not for the Ntrip protocol.

Windows CE is the operating system in the ACU.

With the built-in NtripClient it is easy to set-up the ACU for receiving corrections from a NtripCaster – one only needs an Internet connection. The same mobile phone as above was used for this test system. The communication between the phone and the ACU is in Bluetooth.

The ACU was connected to a Trimble RTK 5700 GPS receiver – that is also operated from the ACU – and calculation of the corrected GPS data is done in SC. Presentation and logging of the data is also taken care of by the SC.

See appendix A for information on how to configure the ACU for Ntrip.

4.5. Latency Measurements

Time latencies of TCP/IP-packets at Internet can be unpredictable and vary from time to time. You can never be sure that a packet will arrive before a predefined time limit, and because latencies are critical for RTCM data this is an important aspect to consider if one wants to send corrections through Internet. Congestion – that is when there is too much traffic at Internet, with the result of reduced throughput or limited possibilities to connect at all – will be a problem as long as Internet works as it does today. These problems can

be reduced though by good, reliable connections and by minimizing the amount of data that is sent.

4.5.1. Equipment and Implementation

Two latency tests have been made for this report. One with a stationary Ethernet line (using the local area network (LAN)), and one with a GPRS connection (using an Ericsson T39 mobile phone with Telia Mobile OnLine/GPRS). The Windows NtripClient provided by BKG was connected to the NtripCaster in BKG's test system in Frankfurt. It received data from the NtripServer that was set up in Gävle, which means that the packets traveled back and forth to Frankfurt.

To monitor the RTCM traffic and its latencies the software "RTCM Control Program for Ntrip GNSS", version 2.0 was used [wBKG]. Latency calculations in the program are made comparing the *modified z-count* in a RTCM message to the internal clock set at local time. Leap seconds (difference between GPS time and GMT time) are software handled, and since the time stamp only has a span of one hour, time zone does not matter. The *modified z-count* has a resolution of 0.6 seconds, therefore the time that is measured is not exact and it is not necessary that the local computer time are absolutely synchronized with the correct time. The local time used in the tests may be wrong by a few tenth of a second.

The total amount of data received was approximately 300 bytes per second. The capacities of both communication methods are more than sufficient to deal with this amount of data.

4.5.2. Results

The latencies were measured 7 of October 2003 between 15.00 and 15.30. This is the time when local network activity is at its peak. No similar information have been available for Telia's GPRS. It is likely that the network down to Germany is relatively busy at this time. Latency measurements have also been performed during night-time in the LAN only. These show similar characteristics as the day-time measurement.

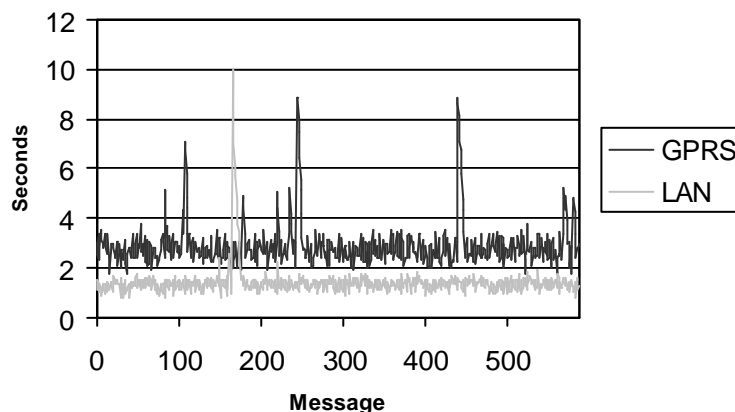


Fig 8. Latency of RTCM type 1 packets

In figure 8 are the approximated latencies in seconds from 10 minutes of traffic. During the tests both RTCM type 1 and RTCM type 31 was sent with a frequency of one per second. Obviously the latency values are almost the same for both types and therefore only the graph for RTCM type 1 is plotted here. Note that these tests have not been performed at the same time (although within the same half an hour of the day), and should not be compared message to message. It is the overall performance that is interesting.

The latency mean of the LAN route is approximately 1.4 seconds and the latency mean of the GPRS route is approximately 2.9 seconds. Note that the difference between the values can be considered rather correct because they have the same error in the time variable. The actual mean can be a few tenth of a second in any direction. The effect of the z-counts resolution is cancelled in the mean.

4.5.3. Geographic Location

When using a local installation of a NtripCaster the packets get a vastly shorter geographic distance to travel. Tests in October 2003 (using ping) demonstrate that the approximate travel time back and forth to the NtripCaster in Germany is about 0.5 seconds. In pure time, this is not much of an improvement (because the travel times are already low), but a more important aspect of a closer located NtripCaster is the reduced risk that something goes wrong during the transportation. Every router the packet has to pass increases the chance of loss or heavy latencies. When using a local NtripCaster only a few routers have to be passed while there are about 15 routers on the way down to the NtripCaster in Frankfurt today.

It is a large project to get statistics of differences in stability of connections depending on location, but one can say that a local NtripCaster is a more reliable option that will generate less latencies.

4.5.4. Latency Differences

An obvious observation is that latency time of the LAN route is less than the GPRS route. This is not a surprise because wireless communication is generally slower and also need to pass more data routers when transforming from GPRS signals to line based Internet traffic. A difference of 1.5 seconds is realistic.

One can also observe that the latencies for the LAN route do not vary as much as the GPRS route. The oscillation of the LAN graph is probably more depending on the resolution of the *modified z-count* than the different packet travel times in the net. This resolution does not fully explain the differences for GPRS – those vary more than 0.6 seconds. One can say that the normal performance – that is when there are no problems in the network – for LAN is more stable than GPRS, but still GPRS seems rather stable and only vary within a one second span, approximately.

LAN is also more secure when trying to avoid peaks with longer latencies. Peaks are usually a result of high amount of data traffic during a short period. There is only one peak with the LAN route while there are a few with the GPRS route. From only ten minutes of traffic one should not make too certain conclusions about this, but it is quite likely that this would be the case for a longer test period too. More things disturb a GPRS packet than a LAN packet, and more frequent longer latencies can therefore be expected.

The latencies are not that bad though. Not even in the worst cases. Over an extended period there would probably be some longer latencies, but as long as several packets in a row does not get these latencies it does not become a problem. A relatively steady stream of correction is the most important goal. One packet now and then can get delayed without decreasing the quality noticeable. That is the case here.

Heavier congestion on the net, or a server that has problems sending its corrections, will happen, but this is hard to prevent and it will probably occur seldom. One way to reduce this is to have short geographic distances between the components.

4.6. Conclusions

This test system shows that it is possible to distribute GPS-data via Internet and that Ntrip is a useful protocol for doing so. Because companies that provide GPS-receivers tends to minimize the users possibilities to make more advanced settings of the instruments it can sometimes be rather cumbersome to make the client/rover working. If Ntrip becomes widely used this will probably not be a problem since the companies will adapt the user interfaces to how their instruments are used in real world implementations.

Given that Ntrip is flexible and open it is relatively easy to build applications that suits special needs. By following the specification the application will be able to communicate with any possible NtripCaster. This is also true for the NtripServer side.

From the latency tests one should not make too certain conclusions, but the given measurements looks promising and they are within acceptable time limits. For more confident conclusions one have to make latency tests under more realistic circumstances.

5. Broadcast corrections

The ongoing Network RTK project in SWEPOS is using VRS to distribute corrections, but there have been interest in other broadcasting techniques too. Broadcast projects have been carried out at Lantmäteriet before (project New RTK). When the idea came up it was rather with FM distribution in mind – since that is one-way communication – but it can be interesting for Internet distribution too. Broadcasting is already used for many applications on Internet.

Test systems, that use different distribution methods, have been developed to get a good approximation of the GPS error in the whole network area. RTCM SAPOS [WB02] is one example of this. Since there is no standardized RTCM message that can handle broadcast yet, the different methods used are hard to overlook. RTCM SC-104 has the task to define a standard for broadcast distribution as a part of the coming RTCM standard. Earliest in the end of 2004 their work is supposed to be ready.

5.1. Broadcast versus Virtual Reference Station

VRS is the most relevant technique to compare broadcast with. They both use several connected reference stations, and they both have one reference station working as a

control center. The difference is in which way the corrections are distributed and what kind of information they contain.

In a Trimble paper [LVC03] the two methods have been compared. It points out that the burden of computation is shifted from the control center to the rover. In real applications of VRS, the computation usually is in the rover; hence these higher demands on the rover should not be a technical problem. For a broadcast system to work one obviously have to distribute more complex software with the rovers, but this also gives more flexibility. Different users may interpret the received correction data in a more free way.

The paper also points out that the drawback of bi-directional communication is not much of a drawback when using communications means such as GPRS. Bi-directional communication is the natural way of communicating in GPRS. In the paper it is stated that it can be an advantage to have two-way communication, which makes it possible to send information about rover availability and different error messages. But one could claim that the authors disregard the fact that there exist certain broadcasting techniques on Internet that probably would work well combined with these techniques.

At the moment in the GPS community there are great interests in different methods for distributing data from one source to multiple clients via Internet – very similar to radio and video streaming – where the messages are multiplied during the transportation. This means that the source only needs to send one copy per message, instead of one single copy to every client as in regular IP traffic. There are a variety of different approaches for this that is usually gathered under the name IP multicasting. A standard is defined [E98], but it depends on one IP address for every multicast that is brought out. This is not a scalable approach and there have been several suggestions – for example [HBH03] and [CDK02] – of application-level based solutions that are scalable and not demand major changes of routers. This is an area that should be considered in future development of broadcasting methods of GPS-data.

VRS and broadcast are also compared in another Trimble paper [TLA02] where it is stated that broadcast can have in principle an unlimited number of users but that it is more complex to add a fee system to it. One needs mechanisms for authorizations to the streams sent out. In VRS billing can come naturally when a rover tries to contact the control center.

Today there exist sophisticated VRS network servers that work together with old receiver software not built for Network RTK. This is not possible in broadcast mode, where rovers must have new software installed. Hence a working broadcast RTK network need to be supported by manufacturers of GPS receivers. Such interests have been shown though.

Another problem with broadcast mode might be that the network messages must be kept fixed during a longer period of time.

An important aspect to consider is that VRS, in general, is better at eliminating both ionospheric and tropospheric errors, compared to broadcasting techniques. The data processor of the rover might be a limit here. One can not use as complicated models as one would like for good precision in the measurements. A discussion of this is out of scope for this paper though.

5.2. RTCM SAPOS

The German AdV (Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland) organization, responsible for the SAPOS reference station network, has introduced a RTCM message that can handle corrections that are sent in broadcast mode. It is called RTCM SAPOS or RTCM type 59 [WB02] – a message type that is reserved for private user messages in the SC-104 standard.

Area correction parameters are the basis of the protocol. In a message, the rover gets these parameters from all the available satellites and can use them in certain given models to calculate its corrections. As with all broadcast distribution the computation work is on the rover.

Message 59 is used in SAPOS network [wSPS]. GPSNet supports SAPOS message and a few of the sources in BKG' s Ntrip test system distribute it.

5.3. Broadcast in GPSNet

GPSNet can be set to broadcast SAPOS messages. When a generator is set to SAPOS-mode it also sends RTCM message 20 and 21 for corrections, and the SAPOS-message are used to increase the RTK performance.

6. Integration in the Present System

An important aspect when discussing possibilities to distribute corrections via Internet is how these new components will work together with the present system running at Lantmäteriet – if Ntrip can be integrated with the present configuration of SWEPOS. One must find out how and where the NtripCaster should be installed, how the corrections are distributed and how the clients/rovers get access to the data.

For demo purposes a local installation of a Ntrip system have been made.

6.1. Local NtripCaster

The NtripCaster should be located somewhere at the control center in Gävle. It makes maintenance easier and the possibilities to integrate with present systems are much improved. Another solution would be to use a NtripCaster installed somewhere else – for example to use BKG' s in Germany – but this would diminish the control from Lantmäteriet. Besides this, a local NtripCaster is a better option for a low and steady latency. Latencies are discussed in section 4.5.

During the project an iGate solution was installed at Lantmäteriet. iGate is a module in GPSTServer [mGS] that can handle communication between sources of RTCM data and the users. It supports the Ntrip protocol and works both as NtripServers, for the sources, and as a NtripCaster. The iGate installation was made available on the public Internet.

A GPSNet RTCM Generator [mGN] was set-up, constantly broadcasting corrections from a reference station in Gävle, via TCP socket communication, to iGate' s IP address on a given port. This generator is handled as a NtripSource, connected internally to a

built-in NtripServer that gets accessible through iGate. The NtripServer-NtripCaster communication is handled internally within iGate.

Via any PC, that has access to Internet, it was then possible to receive corrections with a NtripClient.

It is not necessary to use GPSTServer. A similar system could be set-up, but where other implementations of NtripServers and NtripCaster are used – for example BKG' s system. The socket output from GPSNet can work as NtripSources for other applications.

6.2. RTCM Sources

There are two main approaches considering communication between the control center and SWEPOS stations that works as RTCM sources. One can (1) use Ntrip format during the entire data transportation, or (2) use the already existing network structure and encapsulate RTCM in Ntrip when the data reach the control center.

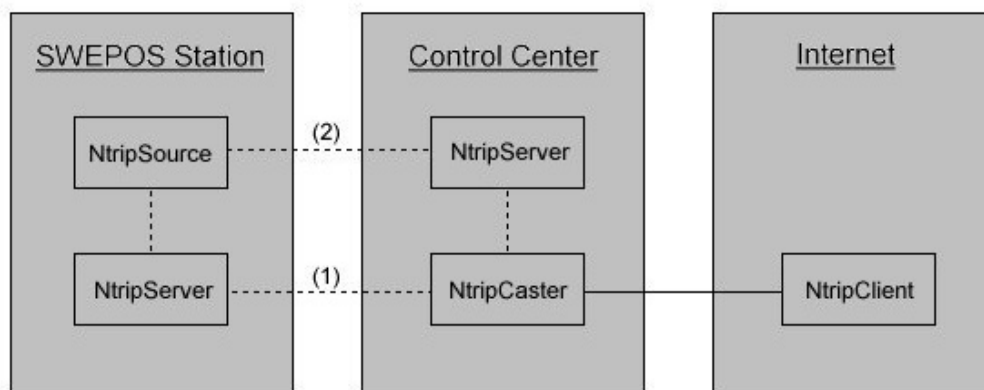


Fig 9. Ntrip communication in SWEPOS

Why the option (2) is to prefer – at least as a first stage – is discussed below.

6.2.1. Ntrip from the source

The first option, to use Ntrip directly from the source, will mean significant modifications in the present configuration. One has to investigate the possibilities to transfer Ntrip over the leased line, – which should be possible without problem – but most important of all, the receiving functionality at the control center has to be modified. The IMoS software needs to be re-placed or re-built, which makes considerable changes of the system necessary.

Small advantages like lesser transformations of messages and a more consistent system can not match the disadvantages of the heavy workload of personnel that would be required. The system would still need to be able to distribute post-processing data, which means that the raw data streams also must be transported with Ntrip. This is possible, but it illustrates some of the major changes that would be necessary in the system.

Problems that can be solved with Internet distribution are mainly rover communication issues. Changing to Ntrip would in most aspects only change the format in which the data are sent from the SWEPOS stations to the control center. No other major advantages would be gained.

6.2.2. Ntrip from sockets

Today RTCM data from the reference stations are available through TCP sockets. Such a stream could be connected to a Command Line NtripServer from BKG [wBKG]; it can handle socket communication. One copy of the application is needed for every station that is connected to the NtripServer.

Another option would be to program a NtripClient that can handle several socket connections. Because all RTCM data streams are physically connected to the same computer this would probably be the best choice. It is easier to maintain and keep control of one single application, instead of one running for each station. The communication in GPSNet works like that, because the software acts both as NtripServers and NtripCaster [mGN]. Also, there would be a relatively easy task to implement a solution that can be suited for the local conditions. Such an application could probably integrate better with IMoS.

This approach is a safer option. The old systems would run just as they did before and Ntrip would work on top of them as an extra feature. If a Ntrip system will be implemented this is probably the best starting point. If this configuration seems to work well, one can start to reflect over more integrated solutions.

6.3. New Rovers

New applications have to be installed, or already used applications have to be re-configured, at the rover side. As described above, Trimble ACU already have support for Ntrip and can work as a NtripClient. Proprietary implementations or BKG's application would work too. Today there exists no Java implementation of a NtripClient. This could be useful because it is the most common language in modern GSM phones. Probably this will become available in the future though. One advantage of Ntrip is its flexibility, thus there is no reason to bind users for a certain NtripClient if customer support reasons can not motivate such demands.

The users only require information about NtripCasters IP-addresses and open port numbers to be able to receive corrections. For billing purposes it is probably interesting to add some sort of authentication to the data streams. The user gets a username and a password, which gives access to streams that have been ordered. This is supported in the Ntrip protocol and is possible to use in for example GPSTest. In GPSTest a database is connected to authentication handling, saving information about the users. More information about authentication can be found in the Ntrip documentation [GW03].

7. Cost and Data Amount

The fee system when using Internet is different from other types of correction distribution systems like GSM or FM. Instead of paying for a usage time or a fixed cost, one pays for the amount of data sent. At least this is true for GPRS. This gives new problems, but also new possibilities. Especially this is essential to Network RTK solutions where large amount of data is sent. Also one has to consider that important changes probably will happen in this area the coming years. The comparison below should be seen as a rough guideline for the situation of today.

7.1. Frequency

Frequency of GPS corrections is usually measured in number of seconds between each message (which means that strictly speaking the name should be period). Naturally, these are fundamental values when discussing cost. Different frequencies can lead to large cost changes. For example, by using only half of these frequencies, the cost per time unit will be halved.

A very common frequency for corrections and observations in real time GPS applications is one per second, while more constant values – like reference station information – are sent with as slow periods as 60 seconds between each message. These are not fixed values, but should be decided considering quality of service and bandwidth demands

7.2. RTCM Size

To be able to calculate the costs of using Ntrip it is necessary to know the size of the packets sent. Since lots of packets will be sent during a regular survey session, the cost will depend not only of the number of packets but also of the size of them. Small differences in packet size may lead to considerable variations of cost.

As mentioned before, the main focus in a Ntrip realization is to use RTCM messages as the correction format. Today RTCM version 2.3 [R01] is the latest officially defined standard, but soon version 3.0 will be finally defined and information about the new standard have already been released in several drafts [R03]. Much of the old, rather cumbersome, treatment of checksums and other bit operations (that was used for other means of communication, such as radio) is removed in the new version, leaving a more simplified protocol that is built on layer principles (see section 2.5). This means that no checksums are included in the layer containing RTCM data in version 3.0. For Internet distribution this is interesting. TCP and IP add a checksum to the packet. When using version 2.3 a lot of redundant information was sent because checksums were used in all layers. Version 3.0 relies on layers below and is able to reduce packet size for a minor decrease in reliability. Also the data fields are completely re-defined in version 3.0, which makes the packets even smaller.

Below is a table showing the different sizes in bytes of some of the RTCM messages of version 2.3 and version 3.0 that might be used in future implementations of a Ntrip system. Generally there is more information in a 3.0 message, thus the sizes should not be compared messages to message between the different versions. In addition to the RTCM documentation calculations are based on RTCM SAPOS [WB02]. Message size of 1001-

1004 should be rounded up to the next integer. For more detailed information about the calculation of RTCM version 2.3 messages in this table, see Appendix B.

RTCM Message	Ver.	Survey	Size (Bytes, where n is #satellites)
1	2.3	DGPS	$10 + \frac{25n + 5(n \bmod 3)}{3}$
3	2.3	DGPS	30
31	2.3	DGPS	$10 + \frac{25n + 5(n \bmod 3)}{3}$
18/20	2.3	RTK	$15 + 10n$
19/21	2.3	RTK	$15 + 10n$
22	2.3	RTK	25
59	2.3	VRS	$10 + \frac{40 + 35n + 5(1 - n \bmod 3)}{3}$
1001	3.0	RTK	$8 + 7.25n$
1004	3.0	RTK	$8 + 15.625n$
1006	3.0	RTK	20
1008	3.0	RTK	6-68

The graph below visualizes the actual packet sizes, with TCP/IP header included [P81a][P81b], of some of the RTCM message packets which sizes are dependent of the number of satellites included in the correction.

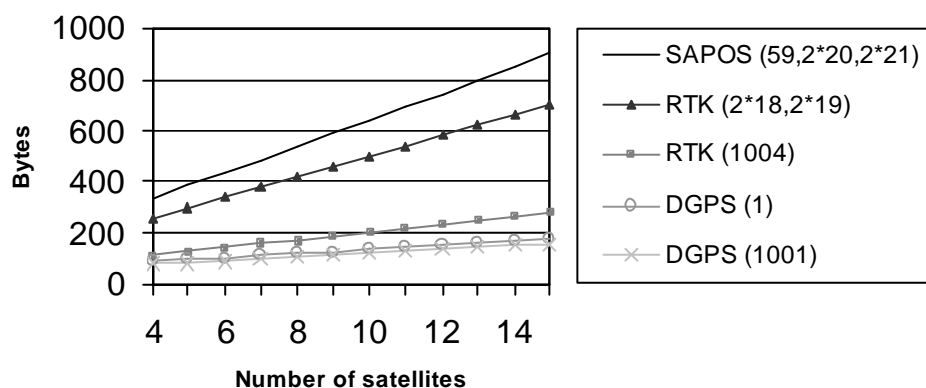


Fig 10. Size of RTCM messages transported in TCP/IP packets

7.3. Cost of GSM and GPRS

The cost to use GPRS is based on the amount of data that is received, while GSM fees are based on usage time. Therefore a comparison can not be done without assumptions about parameters such as size of data packets, data frequencies and time of usage. In figure 11 are four series: (1) a SAPOS messages sent every second, (2) a RTK message in RTCM 2.3 sent every second, (3) a RTK message in RTCM 3.0 sent every second and (4) a GSM connection. Prices are based on present phone contracts at Lantmäteriet (December 2003), and the dial-up fee for GSM is ignored because it is a relatively minor cost. For all

series the assumption is that information from 8 satellites are used in the RTCM messages.

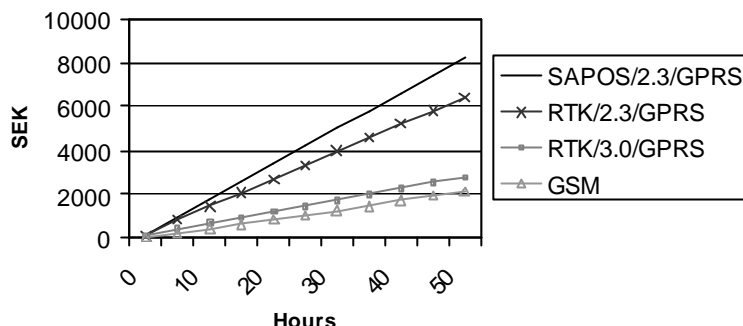


Fig 11. Cost of usage with present telephone fees (December 2003)

An observation is that RTCM 2.3 in combination with GPRS is an expensive option. In RTCM 2.3 it is usually necessary to send more than one type of message at high frequencies, while RTCM 3.0 is more compact and only need one type of message (for example 1004) at high frequency (for example one per second). This means less data and lower GPRS costs. Internet distribution becomes interesting from a cost point of view first when RTCM 3.0 is introduced.

Another observation is that GPRS using the RTCM 3.0 format is slightly more expensive than GSM communication. This is mainly because Lantmäteriet does not have a phone contract that is suited for larger amounts of GPRS data yet. If Lantmäteriet decides to introduce Internet distribution of GPS-data, it will probably be possible to get more profitable contracts.

7.4. Future Considerations

Mobile data distribution is only in its beginning phase, and in the next years there will be big changes in available systems and cost of the present techniques. The major Swedish mobile Internet providers nowadays have special services where fees are based on assumptions that companies will send constant information between different devices. This is relatively new and it is likely that a greater competition about a growing number of customers will force prices down.

Another important aspect to consider is UMTS (Universal Mobile Telecommunication System) – or 3G – technology. It will probably become public available some time during the first half of 2004. In the beginning in limited geographic areas, but if the deployment proceeds as planned, there will be a sufficient cover in Sweden in a few years.

Most likely 3G will be expensive in the beginning, but there will be vast adjustments of prices during the first years. The trend this far seems to be that providers will treat 3G more as a mobile broadband, with none or limited traffic fees and a higher fixed cost. If the fixed cost is at a reasonable rate, this might be a good solution for Internet distribution of GPS corrections. It will be easier to calculate the actual costs in advance, and it will probably be cheaper.

7.5. Conclusions

Mobile data communication is now at a break point. Today one can not motivate a change to Internet distribution based on purely cost aspects, but in a few years it is likely that more competition and flat rates of fees will change this picture. It should be of interest to keep updated in these areas.

Since frequencies are such an important factor when calculating the cost of GPRS distribution, it might be interesting to investigate how often packets have to be sent to keep and accurate positioning. Services where users could decide frequencies of different RTCM-messages could be introduced. It's technically possible to implement a NtripCaster containing information about this, which would lead to services that are more suited for specific users. It would also lead to lower costs.

8. Communication Between NtripCasters

Solutions where Ntrip is used only in parts of the communications can be interesting in the future. One could for example think of systems where a NtripCaster – or some other kind of central unit – is installed locally in an area. Data packets in Ntrip format arrives to the NtripCaster, but is then distributed to the users by some other method of communication. Lantmäteriet would then only distribute data to this NtripCaster, leaving for the local provider to set-up arrangements for the final delivery in for example a radio net.

Another possible scenario is a system with several communicating NtripCasters, where users can access the one closest to them, but still be able to obtain data from other NtripCasters as well. This does not have to be within the same organization. In some countries, like Denmark, there exists a few independent GPS correction providers. A service where customers can access corrections from different providers would probably be of interest.

These and similar tasks would need NtripCasters to be able to communicate with one another – alternatively that it only seems from the user's point of view that they are communicating, while all the processing is done in a NtripClient. In the present configuration of NtripCasters and NtripClients this is not possible. This section will present three suggestions of possible modifications of Ntrip components, for an even more flexible use of the Ntrip protocol. The key aspect is where implementation changes would be required.

8.1. Modified NtripClient

A simple solution to implement, if a user would like to access NtripSources located at different NtripCasters, is to have all the extra processing in a modified NtripClient. This process is invisible for the user. A connection is made to one NtripClient, but the NtripClient merge the source-tables of all available NtripClients and presents a complete list for the user. Connections between the NtripClient and all NtripCasters are direct; the NtripCasters never communicates with one another.

The concept relies on the fact that information about other NtripCasters can be included in the source-table in a data structure named CAS [GW03]. Both IP-address and port number can be stored, and by downloading the source-table from a NtripCaster containing information about another NtripCaster, the NtripClient have all the information needed to access both of them. Internally the NtripClient could, when updating the source-table of one NtripCaster, check for stored information of other NtripCasters. If such information is found a request for a source-table can be sent to the IP address and port number of the other NtripCaster.

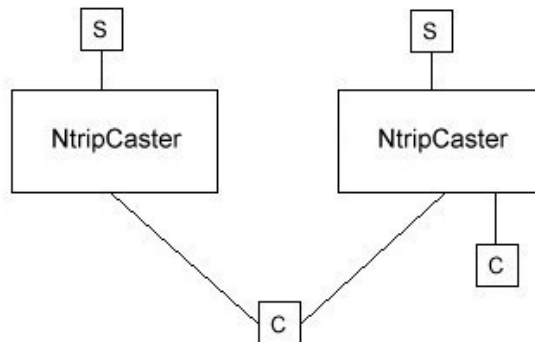


Fig 12. Modified NtripClient

This makes it necessary to change the representation of a NtripSource within a NtripClient. A mountpoint can no longer be seen as unique, since two NtripCasters can be connected to NtripSources with the same mountpoint. Therefore a tuple containing caster identifier and mountpoint have to be used as the unique identifier for each mountpoint. This change is only necessary in the client. The involved NtripCasters does not have to know about other mountpoints.

For this method only NtripClients that wants to access data delivered from several NtripCasters have to be modified. All other system components need no changes and old NtripClients would still work.

This approach may lead to problems when operating in network with security configurations like firewalls. From a communication point of view the NtripClient must have two-way communication directly to all NtripCasters. This makes it impossible to place one NtripCaster in a secured network area only accessed from some other component operated by the correction provider. Solutions presented below will cancel this issue. Also this limits the idea of changing communication channel during the transport. This approach is mainly interesting if one would like to split communication to several NtripCasters on relatively open networks.

8.2. Combined NtripServer and NtripClient

A solution with a bridge-like communication between the NtripCasters can avoid the need to make modifications to any of the existing components. This can be done by using a component that works both like a NtripClient and a NtripServer. The easiest solution is probably to have one for each source on a NtripCaster that should be available on another NtripCaster. It is also possible to merge these into a combined application, where each NtripServer within the application is mapped to a NtripClient.

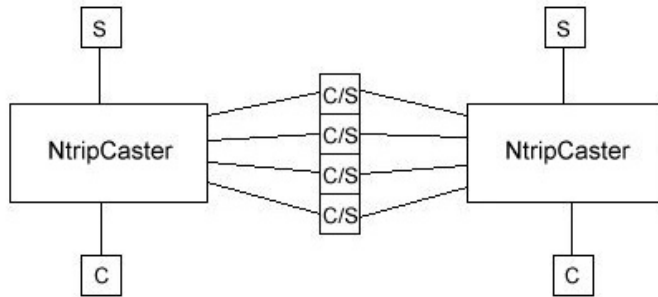


Fig 13. Combined NtripServer and NtripClient

The server part of the combined component is initialized as a normal NtripServer by sending a request to add the mountpoint to the NtripCaster (here called 1) that will distribute the source of another NtripCaster (here called 2). The client part will ask for the source table of NtripCaster 2 and store the mountpoint of interest from NtripClient 2. Observe that the mountpoint of the combined component and the mountpoint of the NtripSource do not have to be the same.

When a request for a RTCM stream arrive, the component activate its client part instead of – as a normal NtripServer would do – starting the stream immediately. It asks NtripCaster 2 for the stream using the stored mountpoint. The stream will be delivered from the NtripSource at NtripCaster 2 and sent to NtripCaster 1, where the requesting NtripClient connected to NtripCaster 1 will fetch it. For every new request of a RTCM stream, a separate client part has to be started.

This methods best suits a network configuration where a central NtripCaster, for example at Lantmäteriet, make NtripSources available on local NtripCasters. For every source that is available, a combined component of this kind is run. It is represented as a NtripServer at the local NtripCaster and as a NtripClient at the central NtripCaster.

8.3. Modified NtripCaster

A third method is to make modifications to the NtripCaster, for example by including NtripClients in the applications.

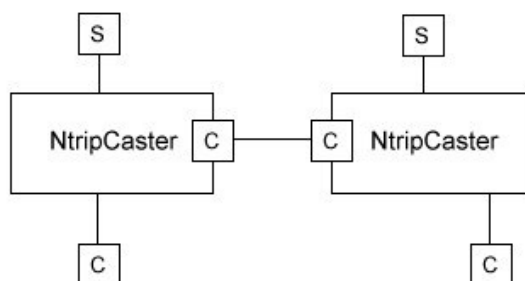


Fig 14. Modified NtripCaster

A request for a stream will trigger a built-in NtripCaster to act as a client at the other NtripCaster. This would have the same security issues as the method in 8.2, where the requesting NtripClient must not have a direct connection to the NtripCaster that handles the source it asks for. The NtripClient does not need to know that the NtripSource is on another NtripCaster either. Everything can be dealt with internally.

The modified NtripCaster have to keep track of which NtripSources that are directly connected and which NtripSources that have to be fetched from another NtripCaster. The communication between the NtripCaster can be closely integrated, where they almost work as one, or on a more independent basis, depending on the demands of the network configuration that will be used.

9. Monitoring the Quality of Corrections

When people call the SWEPOS control center with problems concerning RTCM data distribution, a good tool for the support would be real-time connections to some rovers in the network. The support staff could then see the problem from the user point of view, to determine if there is a problem with the network or with the user's receiver. It would also give constant feedback of the quality of present corrections; and it would be possible to log information for later use, or for statistical calculations.

As an example and evaluation of topics in this paper, the problem was solved with data distribution over a computer network with two communicating components: (1) a logging unit at the position of the rover obtaining data from a GPS-receiver, and (2) a monitoring unit at the control center. The result is an implementation of GPS-data distribution over an intranet.

9.1. Correction Quality Parameters

The true position of the rover is known, hence it is possible to calculate the difference between the true position and the position values that are received from a GPS-receiver using the corrections distributed in the network. For the positioning error the interesting values are (1) difference in latitude, (2) difference in longitude and (3) difference in altitude. The three formulas are:

$$dLat = \frac{lat - lat_0}{180} \cdot r$$

$$dLon = \frac{lon - lon_0}{180} \cdot r \cdot \cos(lat)$$

$$dAlt = alt - alt_0 + N$$

where r is the radius of earth and N is the geoidal separation.

Number of satellites used in the solution, HDOP (Horizontal Dilution Of Precision) and the age of the correction are other values of interest that are received and displayed by the monitoring unit.

Another interesting quality value is the time it takes for the solution to get fixed. This means – simplified – that the ranges from receiver to satellite consist of a fixed integer part and a moving fractional part. In some sense one can say that the calculations have reached an optimal solution then. The process for the solution to get fixed begins every time a new position is to be determined by a rover. This is simulated in the monitor system with a system reset, clearing all known data.

9.2. Components and Communication

Both components of the monitoring system must be physically located to certain positions. The monitor unit has to be at the SWEPOS control center in Gävle, where the support staff is located; and the logging unit has to be located with the GPS-receiver that acts as the rover. The latter was decided to be at a local Lantmäteriet office within the region that one wants to monitor. At these offices the needed data hardware already exists. Also they are connected to the same intranet as the SWEPOS control center, only blocked by a firewall protection.

The control center is located in the S-Net (see figure 15) which is a protected part of LM-net, from where there are direct connections to the network of SWEPOS stations. By opening a port in the firewall between LM-Net and S-Net, dedicated for the one-way communication from the logger to the monitor unit, it is possible to send the data.

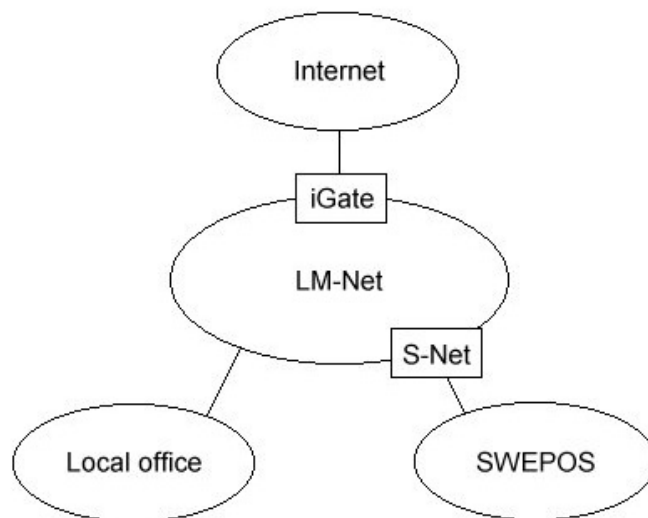


Fig 15. Local network architecture

Another option would have been to use the already installed iGate NtripCaster (see section 6.1). As seen in the figure this would have been a longer and unnecessary packet travel route. One could argue that if many monitor stations would be set up, it would be a good option to have a NtripCaster that could take care of the communication. At the moment only a few monitor stations are planned though. If the concept works well, and there are needs for several more stations, it would probably be better to let the traffic travel through a NtripCaster, especially if one wants to have monitor stations outside the local intranet.

Data sent is in NMEA 0183 GGA format [mZf], which basically is a text string including all the values required. The logger fetches these messages from the GPS-receiver on a constant interval via a serial cable. They are sent with TCP socket communication, on the dedicated port, to the monitor located in S-Net. A special message is sent to the monitor if the GPS-receiver makes a reset. The GGA messages are graphically displayed in the application and logged in text files by the monitor program.

9.3. Implementation

The implementation was done in Borland C++ using libraries for socket communication, serial port communication and graph plotting. Logger and monitor are two independent applications.

The logger application waits for position data (GGA) to be available from a serial port. When data arrives the program check whether it is a fixed solution or not, and increases the proper counters. They are used to count the number of fixed, float or total solutions to be received before a reset command is sent to the GPS-receiver. If it is time for a reset, this command is sent to the GPS-receiver and a message is sent to the monitor, telling about this reset and what time it occurred. A line in the memo windows tells that a message has been received and taken care of by logger.

Communication with the monitor is via TCP sockets, where the logger acts as the client. Fetched GGA messages and created reset messages are sent at this sockets, using the port that has been reserved for this purpose. Both of the sockets are non-blocking so the program will not hang up when waiting for messages to be sent or received. The communication is one-way – only in the direction from the logger to the monitor.

Messages arrived to the socket of the monitor are interpreted in different ways depending on if it is a reset or a GGA message. Reset messages clear the graphs on the screen and set a new time for last reset. GGA messages are dissected on all interesting values, and the values are printed on the screen. Differences are calculated and plotted graphically. Finally the GGA message is logged in a text file, which is named after the present date and hour.

9.4. Local Test

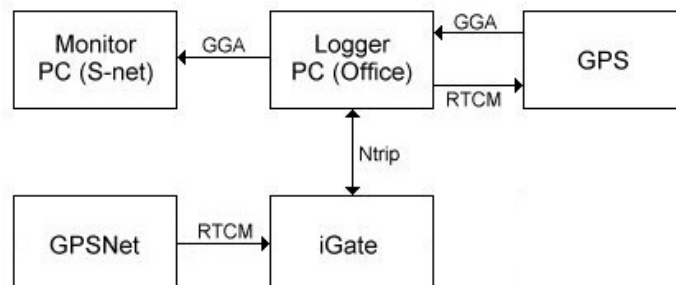


Fig 16. Logger and monitor program connections

The applications were tested on a computer in the SWEPOS control center. Naturally the socket communication between monitor and logger worked without problems – because it only loop back to the same computer –, but there should be no problems separating the components. The socket communication between the two applications was tested in LM-net. Communication between LM-Net and S-Net (see figure 15) should only be a matter of opening one port in the firewall. Figure 16 describes how the system will work. The only difference in this testing is the position of monitor.

An Ashtech GPS-receiver was connected with two serial cables to the computer. One was used for RTCM corrections from a NtripClient – that was running at the computer – and one was used for the GGA messages sent in the other direction. These GGA messages were fetched by logger and sent to monitor. Monitor could then make log files and display quality information on the screen (see figure 17). The RTCM corrections came from a RTCM generator in iGate, located as a virtual reference station outside Gävle.

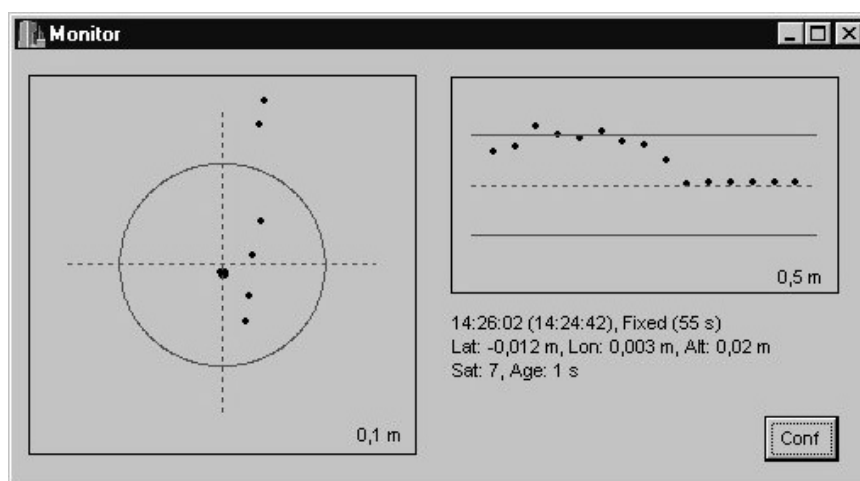


Fig 17. A fixed solution in the monitor

Performance of the quality monitoring was as expected. There were large variations in the time it took to reach a fixed solution, and the solution became better until a fix happened that stabilized the calculations. The time it takes to reach a fixed solution depends to a large extent on the number of satellites and their positions.

10. Conclusions

This paper has shown that distribution of GPS-data via Internet, from SWEPOS, is a possible and interesting communication channel, and that Ntrip is a good method for doing it.

No large hardware investments have to be done. The distribution works with present rover and base station devices, and the only new hardware required is the computer that hosts the NtripCaster, combined with a router and a permanent Internet connection. The communication infrastructure already exists, with GPRS coverage and local networks, which means that nothing have to be built. Instead, more work has to be done on the

software side. One has to make sure that rover software can receive corrections via Internet and that the system at the control center is configured properly – including features as authentication and fee methods. It is possible to either install a completely new system with a stand alone NtripCaster, or use a more integrated approach with the already existing components in GPSNet.

Internet distribution works in both VRS and broadcast mode. New specifications of data formats will adapt better to the characteristics of packet transfer via Internet, while the old formats still work, although they are a bit more cumbersome to use. Broadcasting is an interesting subject for future implementations and a protocol like SAPOS might be useful. A more thorough investigation of broadcast methods should not be performed until the new RTCM broadcast messages are defined by SC-104.

The NtripCaster should be installed locally at Lantmäteriet in Gävle – mostly because of maintenance and integration reasons. Also this will reduce the problem of data packets with long latencies and give more stable connections. The feed of data from reference stations can be handled with Ntrip, but as a first step the best solution would be to maintain the present structure of leased lines, where RTCM streams are fetched from the same sockets as today. Hence the NtripServers will run at the control center and Internet distribution can work together with present systems, making a secure and gradual change to the new system possible.

Several methods for a more flexible use, with more than one NtripCaster, have been suggested. Since Ntrip is an open system, the possibilities are extensive and lies more in the needed workload of implementation than in technical limitations. One should not implement solutions where too many local control centers are needed and the natural non-geographic nature of Internet is ignored. As few steps as possible between user and RTCM source is generally the best approach.

Today a change to Internet distribution would not lead to major cost effects. Just in a few years this picture will probably be completely different, though. New technologies and better availability of mobile Internet will give lower traffic fees. Also the coming new RTCM 3.0 format, with smaller messages, will reduce costs significantly because the common pay method for mobile Internet traffic is based on the amount of data sent. This also gives the possibility to let user pay only for the type of messages and the frequencies that they need; that means better adapted agreements with different customers. The one-time cost to introduce Internet distribution is low.

The test system that was set-up, and the monitor system that was implemented, shows that the performance of Ntrip is as expected. Latencies is not a problem, although more realistic field test would be necessary before it can be claimed for sure. Software and hardware that exists today works with Ntrip, but since it is relatively new functions it sometimes can be rather cumbersome to install; usually the newest software version is required. This is natural with new technologies and will probably be much improved in the future, since companies involved in accurate GPS positioning shows great interests in ongoing projects. An introduction of Internet distribution of GPS-data at Lantmäteriet would probably be successful.

11. References

Articles and books

- [CDK02] Castro M., Druschel P., Kerrmarrec A.-M., Rowstron A., “Scribe: A large-scale and decentralized application-level multicast infrastructure” (2002), *IEEE Journal on Selected Areas in Communications*, Vol. 20, No. 8
- [CTV03] Chen R., Toran-Marti F., Ventura-Traveset J. (2003), “Access to the EGNOS signal in space over mobile-IP”, *GPS Solutions*, Vol. 7, No. 1
- [E83] Ekman M. (1983), *Sfärisk trigonometri och jordellipsoidens geometri*, Lantmäteriet, Gävle
- [E98] Estrin D. et al. (1998), “Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification “, The Internet Engineering Task Force, RFC 2362
- [EZT02] Euler H.-J., Zebhauser B., Townsend B., Wübbena G. (2002), “Comparison of Different Proposals for Reference Station Network Information Distribution Formats”, Proceeding of ION GPS 2002
- [F97] Fielding R. et al. (1997), “Hypertext Transfer Protocol – HTTP/1.1 “, The Internet Engineering Task Force, RFC 2068
- [GW03] Gebhard H., Weber G. (2003), ”Ntrip, Version 1.0”, Federal Agency for Cartography and Geodesy, Germany
- [HBH03] Hjalmtýsson G., Brynjúlfsson B., Helgason O. (2003), “Self-configuring Lightweight Internet Multicast”, Department of Computer Science, Reykjavík University
- [HLC94] Hofmann-Wellenhof B., Lichtenegger H., Collins J. (1994), *GPS, Theory and Practice*, Springer-Verlag, Wien
- [JHW03] Jonsson B., Hedling G., Wiklund P. (2003), “SWEPOS™ Network-RTK Services – status, applications and experiences”, National Land Survey of Sweden
- [LVC02] Landau H., Vollath U., Chen X. (2002), “Virtual Reference Station Systems”, *Journal of Global Positioning Systems*, Vol. 1, No. 2
- [LVC03] Landau H., Vollath U., Chen X. (2003), “Virtual Reference Station versus Broadcast Solutions in Network RTK – Advantages and Limitations”, Trimble Terrasat GmbH
- [P81a] Postel J. ed. (1981),”Internet Protocol”, The Internet Engineering Task Force, RFC 791
- [P81b] Postel J. ed. (1981), “Transmission Control Protocol”, The Internet Engineering Task Force, RFC 793
- [R01] RTCM Special Committee No. 104 (2001), “RTCM Recommended Standards for Differential GNSS (Global Navigation Satellite Systems) Service, Version 2.3”, Radio Technical Commission for Maritime Services

- [R03] RTCM Special Committee No. 104 (2003), “Committee Draft for Vote. RTCM Recommended Standards for Differential GNSS (Global Navigation Satellite Systems) Service, Version 3.0”, Radio Technical Commission for Maritime Services
- [S96] Van Sickle J. (1996), *GPS for Land Surveyors*, Ann Arbor Press, USA
- [TLA02] Talbot N., Lu G., Allison T., Vollath U. (2002), “Broadcast Network RTK – Transmission Standards and Results”, Proceeding of ION GPS 2002
- [WB02] Wübbena G., Bagge A. (2002), “RTCM Message Type 59 – FKP for transmission of FKP, version 1.0”, Geo++

User Manuals

- [mGN] “GPSNet VRS Software User Guide, Version 2.10”, Trimble Geomatics and Engineering Division (2003)
- [mGR] “GRIL Manual, Version 2.3”, Javad Navigation Systems (2003)
- [mGS] “GPSTServer Data Publication Using the Internet User Guide, Version 2.10”, Trimble Geomatics and Engineering Division (2003)
- [mIM] “Operation Manual for IMoS, LMV edition, revision 1.1”, Lantmäteriet (1998)
- [mPV] “PCView User's Manual, Version 2.1.10 MS”, Javad Navigation Systems (2002)
- [mSC] “Trimble Survey Controller Help, Version 10.7”, Trimble (2003)
- [mUR] “UDP Relay, Version 0.9.1”, Geodetic Survey Division, Natural Resources Canada (2003)
- [mZF] “Z-Family GPS Receivers, Technical Reference Manual”, Ashtech, Thales Navigation (2002)

Web Pages

- [wBKG] BKG Ntrip, igs.ifag.de/index_ntrip.htm
- [wIETF] IETF RFC, www.ietf.org/rfc.html
- [wIGS] IGS, igsceb.jpl.nasa.gov
- [wJAV] Javad Navigation Systems, www.javadgps.com
- [wSPS] SAPOS, www.adv-online.de/produkte/sapos.htm
- [wSWEP] SWEPOS, www.swepos.com
- [wTRIM] Trimble, www.trimble.com

Appendix A – Configuration of the built-in NtripClient in Trimble SC

First make sure that version 10.71 (probably later versions will work; at present this is the latest version) is installed.

To begin with you must create a survey style that you can use in the measurements. Click on the icon named *Configuration* in the main window in the Survey Controller, and then choose *Survey styles*. Then, click on the button *New*. A window will pop up where you can fill in the *Style name* (for example “Ntrip”) and *Style type* (choose “GPS”). Then press the *Enter* icon at the bottom right.

Now a window with the configurations for your style type will pop up. Click on *Rover radio*. In the new window, choose “Internet connection” as *type*. New fields will be shown. Fill in the IP-address and the IP-port to the NtripCaster you will be using and mark the *Use Ntrip* check box. Now it should look something like figure A.1 (screen dumps are from Georg Weber at BKG, therefore in German).



Fig A.1. Rover radio



Fig A.2. Rover options

When it looks ok, click on the *Accept* button at the bottom right.

Now choose *Rover options*. This is where you can choose which type of survey you want to perform. RTK is chosen in figure A.2, but naturally this depends on the type of survey and which type of corrections that are sent from the source you are planning to use. When your values are filled in: Click on *Accept*.

Finally store your new survey style by clicking on the *Store* button.

Now your new survey style will be in the list of surveys that can be accessed from the *Survey* icon at the main window.

Appendix B – Size of RTCM 2.3 data packets

This appendix describes the calculation of RTCM 2.3 data packet sizes.

TCP/IP

Ntrip is based on traditional TCP/IP traffic, and therefore the header size of both these protocols have to be included in the calculations. The IP header used in Ntrip communication has the size of 20 bytes, which means that no options in the IP protocol is needed. For TCP the size is also 20 bytes – with no options – leaving for a total of 40 bytes.

Ntrip

The application protocol HTTP is the basis of Ntrip, but there are no overheads of this during the transmission of RTCM packets. The system only has to send actual HTTP/Ntrip messages during initialization of the system. When a connection is made RTCM is the only data that travels with the TCP/IP packet.

RTCM

From the specification of RTCM 2.3 one can get the size in bits of every field in any of the standardized messages. For most of these, simple addition is enough to calculate the size. Some messages have constant size, and some messages depending of number of satellites have easy relations that just add new words for every new satellite. Messages that do not add whole words for each satellite is a bit more complicated to calculate, because of parity and filling. Here follows an example how one can calculate the size in bits of RTCM message 1. Information from the specification gives these values:

$$rtcm_bits = 60 + sat_bits = 60 + 40n + 6r + 8(n \bmod 3),$$

where n is the number of satellites and r is the number of 30 bit words in sat_bits . One can now write the formula

$$sat_bits = 40n + 6 \frac{sat_bits}{30} + 8(n \bmod 3)$$

and by easy calculations, the sat_bits can be defined as

$$sat_bits = 50n + 10(n \bmod 3).$$

Now one can just add 60 to get the total number of bits, $rtcm_bits$.

Finally, for calculation of the actual size in bytes that are sent, one have to consider the “6 of 8” format that RTCM messages are sent in. This means that for every sent byte, six of the eight bits are data, while the most significant bit is always 0 and the second most significant bit is always 1. Since every RTCM word is 30 bits, it takes a total of 5 bytes to send one word. For the calculations this means that we have to add the constant $4/3$ to the total bit value of the message:

$$rtcm_bits_sent = \frac{8}{6}rtcm_bits = \frac{4}{3}rtcm_bits \quad \left(rctm_bytes_sent = \frac{1}{6}rtcm_bits \right)$$

Total

For total size of message, just add together TCP/IP size and RTCM size:

$$total_bytes_sent = tcpip_bytes + rtcn_bytes_sent$$

Verification

To verify that these theoretical values are correct, the RTCM data traffic received to a NtripClient was monitored. It showed to be correct for all messages.

Appendix C – Setting up Logger and Monitor programs

This appendix describes how to set up the logger – connected to an Ashtech receiver – and the monitor. It also describes the possible settings in the programs. RTK is used in this example.

Ashtech Receiver

A source of RTCM corrections for RTK is needed. This can, for example, be achieved with a BKG NtripClient sending corrections via a serial cable to the Ashtech receiver. Some configuration has to be done at the receiver for this. First, define it as a RTK remote station with

```
$PASHS,CPD,MOD,ROV
```

followed by

```
$PASHS,RTC,REM,c
```

where *c* is the serial port where the RTCM corrections will arrive. Then set the baud rate to 9600 at the port where the GGA messages will be sent, using

```
$PASHS,SPD,c,5
```

Observe that you can not use the same port for RTCM and GGA traffic. The frequency in which the GGA messages are sent is set with the command

```
$PASHS,NME,PER,f
```

where *f* is the period in seconds between every message. Logger starts this stream by itself, and turns it off when disconnected. The period value is only needed to be set if the user want to apply something else than the default frequency.

Logger

All settings in Logger are at the start screen. *Monitor IP* and *Monitor Port* are the address where the Monitor program is located. *Fix*, *Float* and *Total* are the limits to a reset should occur – counted in number of GGA messages. *Serial Port* defines the serial port that will receive GGA messages and *Receiver Port* defines the port at the GPS-receiver. Clicking *Connect* makes Logger send a message asking for a GGA stream from the receiver. Logger also resets the GPS-receiver before it starts to read GGA messages.



Fig C.1. Logger

Monitor

To reach the settings in the Monitor, click *Conf*-button and the configuration screen (figure C.2) will appear. *Port* is the listening port of Monitor. When the program starts it is already listening. To change port one need to press *Disconnect* followed by a new *Connect*. The three parameters *Latitude*, *Longitude* and *Altitude* are the position of the GPS antenna that the receiver is connected to. *Earth Radius* is chosen depending on used reference system. The two scale functions at the bottom of the window are for the solid lines displayed in the graphs of the main screen. These set the scale of the display

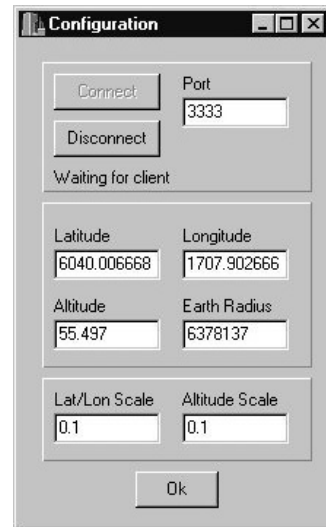


Fig C.2. Configuration

- 1999:2 Andreasson Thor-Björn, Engman Linda: Multipath at the SWEPOS stations
Evaluation of eccosorb, a microwave absorbing material
- 1999:3 Lilje Mikael: Geodesy and Surveying in the future
- 1999:4 Jonsson, Bo: Civil Service Interface Committee International Information
Subcommittee 7 th European Meeting
- 1999:12 Jonsson. Bo: Proceedings of the 13th general meeting of the nordic geodetic
commission.
- 2000:2 Engfeldt Andreas, Jivall Lotti: Kort introduktion till GNSS
- 2000:3 Christina Kempe: Metodstudie för inmätning av skogsbilvägar.
- 2000:4 Niklas Svanholm: Jämförelse av olika metoder att föra över kartdetaljer till ett
nytt koordinatsystem.
- 2000:5 RefStrat – Strategier för referenssystem och referensnät.
- 2001:1 Engberg, Lars: Geodesi 2000 - svensk geodesiverksamhet under kommande
tioårsperiod.
- 2001:6 Jivall, Lotti: SWEREF 99 – New ETRS 89 Coordinates in Sweden, Analysis
Report.
- 2001:7 Jivall, Lidberg, Lilje, Reit: Transformationssamband mellan SWEREF 99 och
RT90/RH 70.
- 2001:10 Rönnberg Andreas: Undersökning av tjänster för differentiell GPS.
- 2002:1 Wiklund Peter (sammanställt rapporten): Slutrapport för projekt Position
Stockholm-Mälaren 1.
- 2002:2 Wahlund Sara: Production measurements with Network RTK tests and
analysis.
- 2002:5 Alfredsson Anders: Studier av deformationer vid byte av koordinatsystem.
- 2002:7 Persson Jan, Brynte Stefan: Kompatibilitet för Nätverks-RTK-programvaran
Trimble GPSNet med olika typer av rörliga mottagare.
- 2003: 4 Engfeldt Andreas, Norin Dan: The 2002 NKG GNSMART/ GPSNet Test
Campaign.
- 2003:10 Engfeldt Andreas, Jivall Lotti: Så fungerar GNSS.

L A N T M Ä T E R I E T



Blankettförrådet 801 82 GÅVLE Tfn 026 - 63 30 00 Fax 026 - 68 75 94
Internet www.lantmateriet.se