

Rapporti tecnici

INGV

**GPSView: monitoraggio in tempo reale
delle stazioni GPS acquisite tramite
Nanometrics Libra VSAT**

270



Editorial Board

Andrea Tertulliani - Editor in Chief (INGV - RM1)
Luigi Cucci (INGV - RM1)
Nicola Pagliuca (INGV - RM1)
Umberto Sciacca (INGV - RM1)
Alessandro Settimi (INGV - RM2)
Aldo Winkler (INGV - RM2)
Salvatore Stramondo (INGV - CNT)
Gaetano Zonno (INGV - MI)
Viviana Castelli (INGV - BO)
Marcello Vichi (INGV - BO)
Sara Barsotti (INGV - PI)
Mario Castellano (INGV - NA)
Mauro Di Vito (INGV - NA)
Raffaele Azzaro (INGV - CT)
Rosa Anna Corsaro (INGV - CT)
Mario Mattia (INGV - CT)
Marcello Liotta (Seconda Università di Napoli, INGV - PA)

Segreteria di Redazione

Francesca Di Stefano
Tel. +39 06 51860068
Fax +39 06 36915617
Rossella Celi
Tel. +39 095 7165851
redazionecen@ingv.it



Rapporti tecnici

INGV

GPSVIEW: MONITORAGGIO IN TEMPO REALE DELLE STAZIONI GPS ACQUISITE TRAMITE NANOMETRICS LIBRA VSAT

Luigi Falco

INGV (Istituto Nazionale di Geofisica e Vulcanologia, Centro Nazionale Terremoti)

270

Indice

Introduzione.....	7
1. Uno sguardo al ricetrasmittitore satellitare Cygnus.....	7
2. NMX Protocol	8
3. NaqsServer.....	9
4. GPSView	10
4.1. GPSView Communication Protocol.....	11
4.2. Connect Message.....	11
4.3. Channel List.....	12
4.4. AddSerialChannel.....	12
4.5. Data Message.....	13
4.6. Terminate Subscription.....	13
5. Interfaccia utente GPSView	13
Conclusioni e sviluppi futuri	15
Bibliografia	15

Introduzione

Libra VSAT di Nanometrics [Nanometrics, 2003] è uno dei sistemi di telemetria satellitare utilizzato in INGV per la trasmissione dei dati sismici [Falco, 2006]. Ogni stazione remota è costituita da un ricetrasmittente satellitare *Cygnus* che archivia i segnali sismici provenienti dal digitalizzatore *Trident* e li invia via satellite ai server di acquisizione dati *NaqServer* dislocati in diverse sedi dell'INGV. Il *Cygnus* prevede, inoltre, la possibilità di acquisire un flusso dati seriale; tale collegamento è ampiamente utilizzato in INGV per l'acquisizione del flusso dati seriale proveniente da ricevitori GPS della Rete Integrata Nazionale GPS (RING). I *NaqServer*, quindi, ricevono in tempo reale il flusso dati sismico e il flusso dati GPS tramite il ricetrasmittente satellitare *Carina*. Affinché la RING possa garantire elevati standard di efficienza è opportuno poter monitorare costantemente lo stato delle acquisizioni di tutte le stazioni GPS della rete al fine di intervenire rapidamente su eventuali guasti o malfunzionamenti. Per le stazioni che utilizzano collegamenti INTERNET, cellulare o WiFi esistono appositi tool che consentono queste verifiche; per le stazioni che invece utilizzano il sistema di acquisizione Libra VSAT è stato sviluppato GPSView che consente di monitorare in tempo reale i flussi seriali (dati GPS) delle stazioni di monitoraggio sismico. La Fig. 1 mostra il layout semplificato dell'infrastruttura Nanometrics Libra VSAT in uso presso l'INGV.

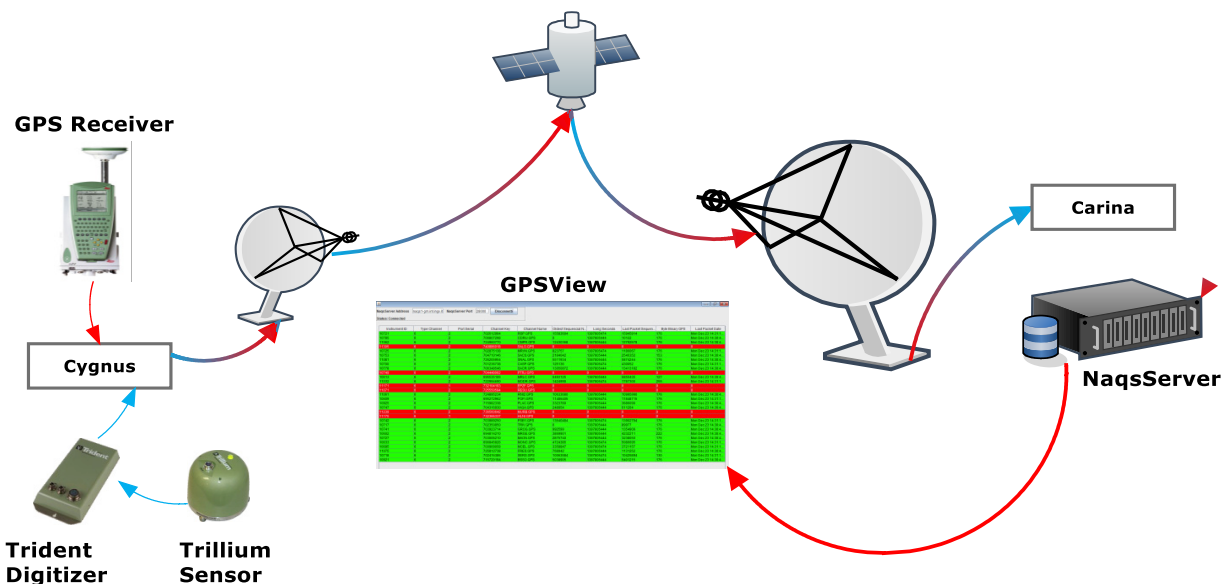


Figura 1. Layout semplificato dell'infrastruttura di acquisizione dati Nanometrics Libra VSAT presso l'INGV.

1. Uno sguardo al ricetrasmittente satellitare Cygnus

Il ricetrasmittente satellitare *Cygnus* è installato presso ogni stazione sismica remota della rete Libra VSAT; esso si presenta come in Fig. 2.



Figura 2. Ricetrasmittitore satellitare.

Nella parte anteriore sono visibili le seguenti connessioni: 2 porte seriali, 2 porte NMX, connettore di alimentazione, porta RX e TX per la ricezione e la trasmissione satellitare, connettore sensore di temperatura e connettore GPS (per il timing interno da non confondere con i ricevitori GPS della RING). Le porte NMX collegano il Cygnus al digitalizzatore Trident tramite un collegamento seriale proprietario denominato *NMXbus*. Questo tipo di collegamento trasporta esclusivamente dati sismici. Le 2 porte seriali, identificabili dall'etichetta *COMMS*, sono utilizzate, invece, per il collegamento del ricevitore GPS (della RING) al Cygnus. La configurazione tipica in uso presso l'INGV prevede: una porta seriale configurata in modalità *Interactive* che consente di comunicare col ricevitore GPS per inviargli, ad esempio, parametri di configurazione; la seconda, invece, configurata in modalità *Serial Receive* per ricevere il flusso dati del ricevitore GPS; è proprio questo flusso dati che giunge ai server di acquisizione NaqsServer e viene analizzato dal tool GPSView; è comunque possibile utilizzare ricevitori GPS con un'unica porta seriale e continuare a monitorare con GPSView il loro flusso dati.

2. NMX Protocol

I dati ricevuti sulla porta seriale (quella configurata in modalità *Serial Receive*) del ricetrasmittitore satellitare Cygnus sono pacchettizzati nel formato NMXP ed incapsulati nei pacchetti standard UDP prima di essere trasmessi via satellite. Il protocollo NMX consente di trasportare, oltre i dati sismici, anche quelli GPS acquisiti tramite flusso seriale, dallo strumento remoto fino ai NaqsServer presso le sedi INGV.

Le comunicazioni tra i Cygnus remoti ed i server Naqs sono completamente *stateless* perché basate su UDP. Se da un lato il protocollo UDP vincola il NaqsServer a farsi carico di eventuali richieste di ritrasmissione dati va tenuto presente che consente anche un notevole risparmio di banda satellitare rispetto all'overhead introdotto da protocolli di trasporto "orientati alla connessione" come TCP.

Ogni pacchetto dati (sismico o GPS) è etichettato con uno specifico "canale-sequential number" e fornisce inoltre informazioni riguardo il sequential number del pacchetto dati più vecchio disponibile per quel canale. Il server Naqs, quindi, può richiedere la ritrasmissione di determinati pacchetti persi o corrotti identificandoli tramite il loro sequence-number. Il Cygnus riceve tale richiesta e reinvia i pacchetti specificati non appena la banda satellitare lo consente.

3. NaqsServer

Il NaqsServer è il server di acquisizione dati progettato per ricevere, processare ed archiviare i dati sismici e seriali provenienti dalle stazioni remote. NaqsServer usa una architettura di tipo “channel-oriented”; i dati di ogni canale (un canale identifica una componente sismica o un flusso seriale GPS) sono etichettati, temporizzati e compressi dai Cygnus remoti ed inviati ai Naqs in pacchetti di dimensioni fisse. Siccome i pacchetti vengono etichettati dai Cygnus che li generano il Naqs usa questa informazione per identificare il mittente di un determinato dato ed archivarlo localmente nel relativo canale. Lo scopo primario del server Naqs è quello di collezionare e archiviare i dati in ingresso nei ringbuffer (memorie a coda circolare) associati a ciascun canale affinché possano essere estratti e processati in un secondo momento. Il Naqs assicura inoltre un buon livello di completezza dei dati effettuando automaticamente richieste di ritrasmissione per ogni pacchetto perso o corrotto durante la trasmissione.

Altra fondamentale caratteristica di NaqsServer è quella di fornire l’accesso alle forme d’onda e ai dati seriali (GPS nel nostro caso) attraverso delle sottoscrizioni TCP. Lo *Stream/System Manager*, un componente software dell’architettura del NaqsServer, infatti, agisce come un Data Server; accetta le connessioni e le richieste dai vari client e fornisce loro i vari flussi dati in tempo quasi-reale. La Fig. 3 mostra l’architettura software del NaqsServer.

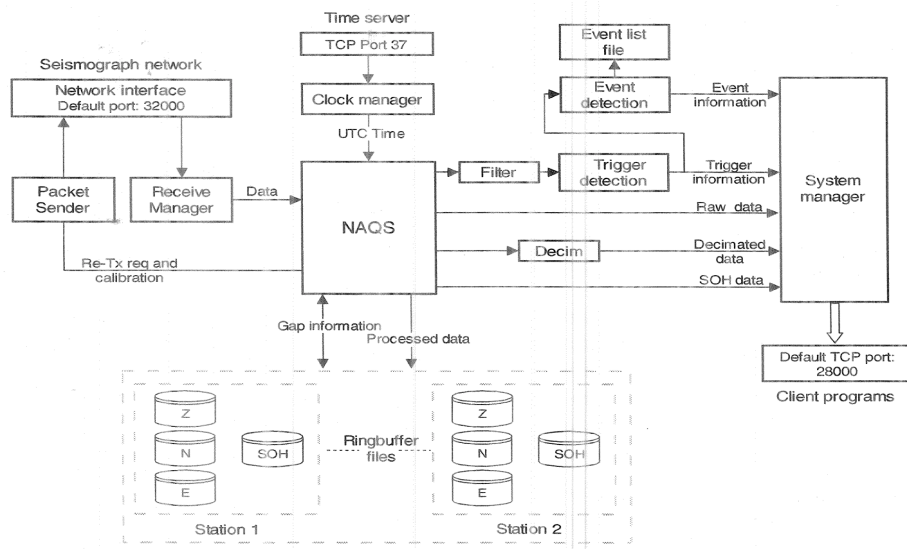


Figura 3. Architettura software NaqsServer.

Le forme d’onda delle stazioni sismiche possono essere richieste nel formato compresso o non compresso mentre i flussi dati seriali sono sempre trasmessi in formato compresso. I dati compressi, inoltre, possono essere richiesti nelle seguenti modalità:

- Raw Stream: tutti i pacchetti (sia originali che ritrasmessi) vengono inviati ai client nello stesso ordine in cui sono stati ricevuti via satellite dal server Naqs. In questo caso i pacchetti possono essere mancanti, duplicati o non in ordine ma vengono ricevuti con una latenza minima
- Buffered Stream: I pacchetti di ciascun canale vengono inviati ai client in ordine cronologico; eventuali gap vengono prima colmati da pacchetti ritrasmessi.

Così come il client *Waveform* (visualizzatore di forme d’onda fornito col sistema Nanometrics Libra VSAT) si connette al Naqs e ne richiede le forme d’onda allo stesso modo GPSView ne richiede il flusso dati GPS al fine di verificarne le latenze.

4. GPSView

GPSView è scritto in Java; in particolare utilizza la libreria `java.net` per la gestione delle `socket` di connessione verso il server Naqs e `java.swing` per la realizzazione dell'interfaccia utente. Per il suo corretto funzionamento richiede il set di librerie runtime *Java Runtime Environment* che essendo disponibili per diverse architetture hardware e software garantisce a GPSView una elevata portabilità. L'ambiente di sviluppo (IDE) utilizzato è stato NetBeans, progetto open source attualmente sponsorizzato da Oracle. La fig.4 mostra l'architettura software di GpsView; in particolare le classi Java realizzate che utilizzano il *Communication Protocol* di Nanometrics descritto nel paragrafo successivo.

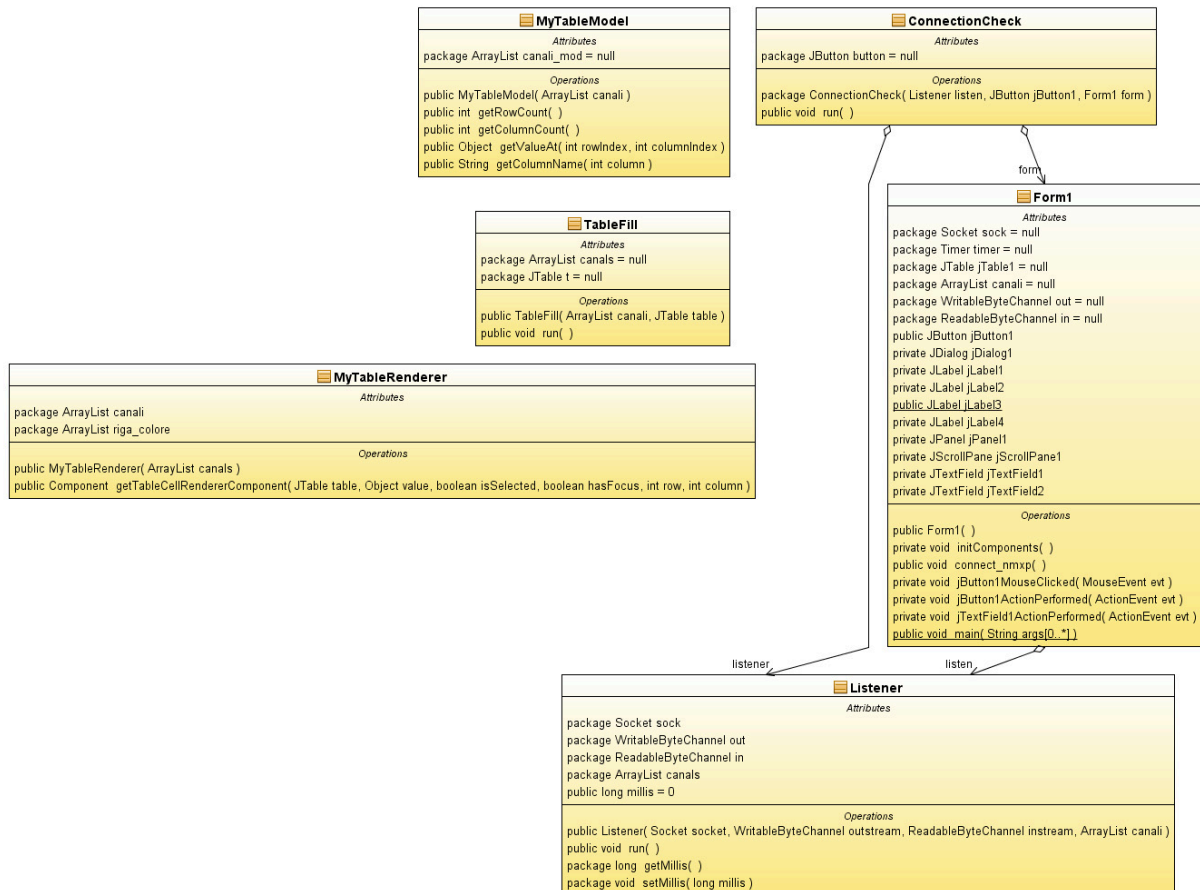


Figura 4. Architettura software GPSView.

L'utente interagisce con la classe `Form1` che realizza l'interfaccia grafica di GPSView. Tra gli attributi di questa classe sono infatti presenti il pulsante di connessione, la casella di testo nella quale andrà specificato l'indirizzo ip del NaqsServer a cui connettersi e la tabella che mostra le latenze per ogni singolo canale. Tra i metodi, invece, va evidenziato il `connect_nmxp()` che realizza i punti 1, 2, 3 e 4 del *Communication Protocol* descritto nel paragrafo successivo consentendo a GPSView di connettersi al NaqsServer e sottoscrivere i canali GPS che si intende monitorare.

La classe `Listener` si occupa, invece, della ricezione dei dati una volta che la connessione al NaqsServer è avvenuta con successo; implementa sostanzialmente il punto 5 del *Communication Protocol* e si preoccupa, inoltre, di disegnare la tabella di visualizzazione delle latenze secondo il modello descritto dalla classe `MyTableModel`, popolandola dei valori ricevuti dal NaqsServer attraverso la classe `TableFill` ed associando diverse colorazioni alle righe della tabella a seconda delle latenze calcolate utilizzando la classe `MyTableRenderer`.

Infine, alla classe *ConnectionCheck* è affidato il compito di monitorare costantemente la connessione di GPSView verso il NaqsServer ripristinandola automaticamente in caso di interruzione.

4.1. GPSView Communication Protocol

Ogni client software, così come GPSView, deve utilizzare, quindi, uno specifico protocollo di comunicazione, denominato *Communication Protocol* [Nanometrics, 2003], affinché sia possibile richiedere al Naqs i flussi dati che si intende acquisire. Il *Communication Protocol* è schematizzabile nei seguenti punti (in corsivo sono indicati i tipi specifici di messaggi che saranno approfonditi successivamente):

1. Apertura di una socket verso lo *Stream/System Manager* di Naqs utilizzando la porta specificata nella configurazione del Naqs (di default è la 28000);
2. Invio di un messaggio *Connect* allo *Stream/System Manager*;
3. Ricezione di una lista di canali (*Channel List*) dallo *Stream/System Manager*. Questa è la lista di tutti i canali sismici e seriali disponibili sul server Naqs;
4. Invio del messaggio *AddSerialChannel* nel quale si specificano al Naqs tutti i flussi dati che si intende acquisire;
5. Ricezione del messaggio *Data messages* contenente i flussi in tempo reali dei canali sottoscritti al punto 4;
6. Invio del messaggio di *Terminate Subscription* quando si intende chiudere la connessione col Naqs;
7. Chiusura della socket.

La Fig. 5 mostra gli step del Communication Protocol

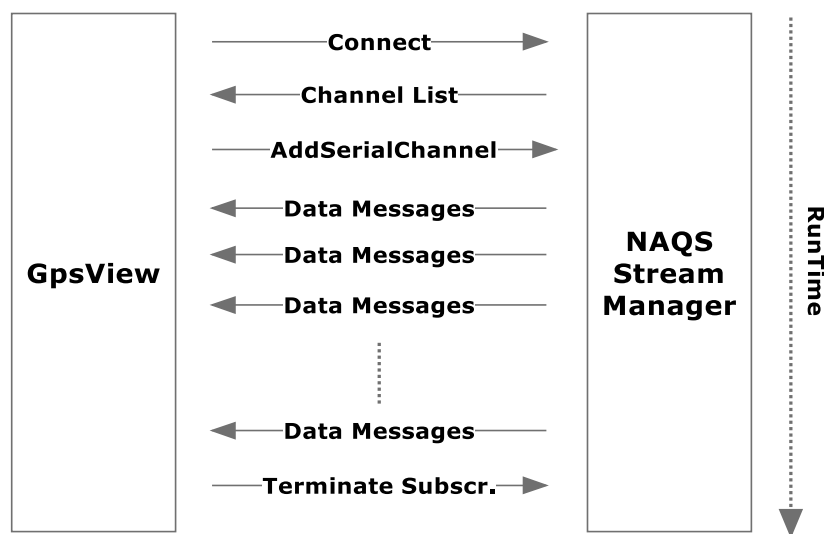


Figura 5. Schema Communication Protocol.

Di seguito le specifiche dei messaggi del *Communication Protocol*.

4.2. Connect Message

Immediatamente dopo l'apertura della socket verso il Naqs il *Communication Protocol* prevede l'invio del messaggio *Connect*. Di seguito il suo formato:

Header	
4 byte int	Signature = 0x7ABCDE0F
4 byte int	Message type = 100
4 byte int	Message content length = 0

Ogni messaggio è sempre preceduto da una *Signature*; il suo scopo è quello di comunicare allo *Stream/System Manager* che i successivi Byte del pacchetto dati rappresentano gli specifici messaggi del Communication Protocol.

4.3. Channel List

Alla richiesta di connessione da parte di GPSView il Naqs risponde con la lista di canali disponibili utilizzando il messaggio *Channel List* così composto:

Header		
4 byte int	Signature = 0x7ABCDE0F	
4 byte int	Message type = 150	
4 byte int	Message content length = 4 + N * 16	
Content		
4 byte int	Number of channel available = N	
N * 16 byte info bundle	4 byte int	Channel Key = (ID << 16) (type << 8) channel) Where ID is the full instrument serial number Type is the type subtype: 1 = time series 2 = state of health 6 = transparent serial Channel is the data channel number (0 to 5)
	12 byte int	Zero-terminated channel name string (e.g. STN01.BHZ)

Questo tipo di messaggio consente al client di verificare quali canali dati sono disponibili sul NaqsServer e fruibili in tempo reale attraverso lo *Stream/System Manager*. Come riportato in precedenza il client Nanometrics Waveform, visualizzatore di forme d'onda, nella fase successiva alla ricezione del messaggio *ChannelList* effettuerà la sottoscrizione del subtype pari a 1 che corrisponde alle serie temporali (sismogrammi). GPSView, invece, che si propone di analizzare il contenuto del flusso dati GPS acquisito tramite collegamento seriale, effettuerà la sottoscrizione del subtype pari a 6 attraverso l'invio allo *Stream/System Manager* di un messaggio denominato *AddSerialChannel* e descritto nel paragrafo successivo.

4.4. AddSerialChannel

Una volta ottenuta la lista di canali GPSView sottoscrive solo quelli di cui vuole acquisirne il flusso dati ed in particolare quelli di tipo "6" cioè "transparent serial". A tal proposito invia al Naqs il messaggio *AddSerialChannel* nella seguente forma:

Header	
4 byte int	Signature = 0x7ABCDE0F
4 byte int	Message type = 124
4 byte int	Message content length = 12 + 4 * N, where N = number of channels
Content	
4 byte int	Number of channel requested = N (use 0 to request all channels)
N * 4 byte int	Channel Key for each requested channel (same as in Channel List message)
4 byte int	Short-term-completion time = s, -1 <= s <= 300 seconds (-1 indicates no short-term completion)
4 byte int	Buffer flag 0 = do not send buffered packets for these channels 1 = send buffered packets for these channels

Quando si verifica una perdita di pacchetti nella trasmissione dati dal sito remoto al server di acquisizione NaqsServer, lo *Stream/System Manager* attenderà un tempo configurabile (*Short-term-completion time*) affinché il gap venga colmato da eventuali ritrasmissioni ed il client possa ricevere i dati in ordine cronologico. Se si specifica un valore di *Short-term-completion time* pari a 0 significa che i dati saranno inviati al client (GPSView) in ordine cronologico, senza però attendere eventuali dati persi durante la trasmissione; un valore di *Short-term-completion time* pari a -1, invece, consentirà allo *Stream/System Manager* di inviare al client (GPSView) i pacchetti dati non appena sono disponibili anche se non in ordine cronologico. Nello specifico dell'implementazione di GPSView si è scelto di impostare questo valore a 0 in quanto lo scopo principale del software non è l'analisi del contenuto informativo del pacchetto seriale ma la verifica di funzionamento della trasmissione dati del ricevitore GPS. Ovviamente in caso di *Short-term-completion time* maggiore di 0 è indispensabile impostare anche il *Buffer Flag* a 1 in quanto nell'attesa di ritrasmissioni il Naqs colleziona momentaneamente i dati da inviare in appositi buffer.

4.5. Data Message

Una volta terminata la sottoscrizione dei canali il Naqs comincerà ad inoltrare a GPSView il flusso dati dei canali sottoscritti. I messaggi che GPSView riceverà avranno il seguente formato:

Header	
4 byte int	Signature = 0x7ABCDE0F
4 byte int	Data type = 1
4 byte int	Data Content Length = variable
Content	
4 byte int	Oldest sequence number
N byte	N byte compressed data packet

4.6. Terminate Subscription

In ultimo, quando GPSView intenderà chiudere la comunicazione col server Naqs gli invierà il messaggio di *Terminate Subscription* specificando eventualmente la ragione della disconnessione:

Header	
4 byte int	Signature = 0x7ABCDE0F
4 byte int	Data type = 200
4 byte int	Data Content Length = 4 + N
Content	
4 byte int	Reason for termination: 1 = Normal shutdown 2 = Error Shutdown 3 = Timeout shutdown
N byte string	String message (none if N = 0)

5. Interfaccia utente GPSView

L'utilizzo di GPSView è estremamente semplice; una volta avviato occorre inserire l'hostname e la porta tcp del server Naqs a cui connettersi e del quale si desidera monitorare il flusso dati delle stazioni GPS.

Instrument ID	Type Channel	Port Serial	Channel Key	Channel Name	Oldest Sequential N.	Long Seconds	Last Packet Sequen...	Byte Binary GPS	Last Packet Date
11278	6	2	739116546	SSY.GPS	0	0	0	0	
10608	6	2	695207426	CERT.GPS	2511051	1392374144	2873361	176	Fri Feb 14 11:35:44 ...
10738	6	2	703727106	SALB.GPS	15615573	1392374534	15977883	176	Fri Feb 14 11:42:14 ...
10647	6	2	697763330	HVZN.GPS	0	1392374534	223034	176	Fri Feb 14 11:42:14 ...
11017	6	2	722011650	MTSN.GPS	4583797	1392374534	4946107	176	Fri Feb 14 11:42:14 ...
10746	6	2	704251394	SGRT.GPS	16134611	1392374534	16496921	176	Fri Feb 14 11:42:14 ...
10720	6	2	702547458	NOCI.GPS	809583	1392374534	1171893	176	Fri Feb 14 11:42:14 ...
10748	6	2	704382466	SCTE.GPS	5531240	1392374414	5893550	176	Fri Feb 14 11:40:14 ...
10762	6	2	705299970	MSAG.GPS	16096988	1392374534	16489298	176	Fri Feb 14 11:42:14 ...
11201	6	1	734070273	NOV.GPS	455904	1392374534	818214	176	Fri Feb 14 11:42:14 ...
11306	6	2	740951554	RAFF.GPS	0	0	0	0	
11174	6	1	732300801	EFIU.GPS	0	0	0	0	
10893	6	2	700777988	PTRJ.GPS	0	1392374534	83144	176	Fri Feb 14 11:42:14 ...
10708	6	1	701761025	ILLI.GPS	0	0	0	0	
11028	6	2	722732546	ROVR.GPS	1033887	1392374144	1396197	176	Fri Feb 14 11:35:44 ...
10495	6	2	687801858	PALZ.GPS	184946	1392374534	254418	176	Fri Feb 14 11:42:14 ...
10884	6	1	711984641	IFIL.GPS	140935471	1392374534	141297781	135	Fri Feb 14 11:42:14 ...
11171	6	1	732104193	EPZF.GPS	0	0	0	0	
11039	6	2	723453442	CELI.GPS	9037863	1392374534	9400173	176	Fri Feb 14 11:42:14 ...
10919	6	2	715589122	CERA.GPS	1889878	1392374504	2252188	176	Fri Feb 14 11:41:44 ...
11068	6	2	725353986	SIRI.GPS	2602560	1392374534	2964870	176	Fri Feb 14 11:42:14 ...
20626	6	1	1351747073	HMDC.GPS	0	1392374534	141069	176	Fri Feb 14 11:42:14 ...
10791	6	2	707200514	MTRZ.GPS	1707298	1392374534	2069608	176	Fri Feb 14 11:42:14 ...
11185	6	1	733021697	PTMD.GPS	0	0	0	0	
10853	6	1	711263745	ECNV.GPS	0	0	0	0	
11079	6	2	726074882	MTTG.GPS	0	0	0	0	
11204	6	1	734266881	FAVR.GPS	5749119	1392374534	6111429	255	Fri Feb 14 11:42:14 ...
10726	6	2	702940674	MCRV.GPS	52549	1392374534	414859	176	Fri Feb 14 11:42:14 ...
20607	6	1	1350501889	HAGA.GPS	0	0	0	0	
20647	6	1	1353123329	HSRS.GPS	0	0	0	0	
10719	6	2	702481922	CAFI.GPS	5800225	1392374504	6162535	176	Fri Feb 14 11:41:44 ...
11064	6	1	725091841	ECHR.GPS	0	0	0	0	
20634	6	1	1352271361	HAVL.GPS	11776676	1392374534	12138986	176	Fri Feb 14 11:42:14 ...
10639	6	1	697239041	MPNC.GPS	8854744	1392374504	9217054	222	Fri Feb 14 11:41:44 ...
11170	6	1	732038657	MCSR.GPS	6673687	1392374504	7035997	104	Fri Feb 14 11:41:44 ...
11007	6	2	721356290	HCRL.GPS	0	0	0	0	
10752	6	1	704644609	IACL.GPS	0	1392374534	1707	135	Fri Feb 14 11:42:14 ...
11242	6	1	736757249	IMCR.GPS	5464218	1392374534	5826525	176	Fri Feb 14 11:42:14 ...
20664	6	1	1354237441	EMCN.GPS	0	0	0	0	
10703	6	2	701493346	INPP.GPS	4436316	1392374504	4788826	176	Fri Feb 14 11:41:44 ...

Figura 6. Interfaccia utente GPSView.

La Fig. 6 mostra come si presenta l'interfaccia di GPSView a connessione avvenuta. Ogni riga della tabella rappresenta un canale GPS acquisito dal Naqs a cui si è connessi. Per ciascun canale vengono visualizzate le seguenti informazioni:

Campo	Descrizione
Instrument ID	Serial number completo del Cygnus remoto
Type Channel	Tipo di flusso dati proveniente dal Cygnus; 6 in caso di pacchetti seriali (Il ricevitore GPS è infatti collegato al Cygnus tramite collegamento seriale)
Port Serial	Il Cygnus dispone di 2 porte seriali; Generalmente la 1 viene utilizzata per interagire con il ricevitore GPS (<i>Interactive mode</i>) la 2 per il flusso dati (<i>Serial Receive</i>); questo parametro consente di stabilire a quale porta è collegato il flusso dati
Channel Key	È l'identificativo univoco del canale
Channel Name	Nome mnemonico identificativo del canale
Oldest Sequential Number	Indica il sequence number più vecchio presente nei buffer del Naqs
Long Seconds	Tempo espresso in secondi dal 1 gennaio 1970 dell'ultimo dato ricevuto (su questo parametro viene effettuato il controllo delle latenze)
Last Packet Sequential Number	Indica il sequence number dell'ultimo pacchetto acquisito dal Naqs
Byte Binary GPS	Indica il numero di byte grezzi (RAW) incapsulati nel pacchetto NMX del flusso dati GPS
Last Packet Date	Esprime lo stesso valore di Long Seconds ma nel formato comprensibile all'uomo

Il controllo in tempo reale dei flussi dati consente inoltre di monitorare le latenze per ciascun canale e rendere immediatamente evidente all'utente lo stato delle acquisizioni per ciascuna stazione GPS.

In particolare ogni riga della tabella (e quindi ogni canale) può subire 3 diversi tipi di colorazione a seconda che si verifichi:

- una latenza inferiore a 2 minuti (**VERDE**)
- latenza compresa tra 2 minuti e 10 minuti (**ARANCIONE**)
- latenza superiore a 10 minuti o stazione guasta (**ROSSO**)

Conclusioni e sviluppi futuri

GPSView nasce fondamentalmente per consentire agli operatori della Rete Integrata Nazionale GPS (RING) di verificare in tempo reale lo stato delle acquisizioni delle stazioni che utilizzano il vettore di trasmissione dati Nanometrics Libra VSAT.

GPSView potrebbe essere notevolmente migliorato con l'aggiunta di ulteriori funzionalità. Allo stato attuale, ad esempio, non è possibile storicizzare i valori di latenza registrati durante l'esecuzione del software. Sarebbe interessante, in tal senso, produrre per ciascuna stazione GPS appositi grafici che consentano di graficare il parametro latenza nel tempo.

Il contenuto informativo del dato seriale proveniente dal ricevitore GPS viene attualmente utilizzato solo per verificare lo stato di acquisizione della stazione remota e non viene in alcun modo elaborato per produrre ad esempio parametri di qualità o valori di posizionamento in tempo reale. Di sicuro rappresenterebbe uno sviluppo futuro molto interessante.

Infine, oltre a fornire funzionalità client verso il server Naqs, GPSView potrebbe prevedere un modulo server con relativo *Stream Manager* al quale i software di elaborazione dati più evoluti potrebbero connettersi in tempo reale.

Bibliografia

- Falco Luigi (2006). *Realizzazione rete di acquisizione dati Nanometrics e segmento PDMZ (Partial DeMilitarized Zone) della rete telematica della sede di Grottaminarda dell'Istituto Nazionale di Geofisica Vulcanologia*. Rapporti Tecnici INGV. Numero 35.
- Nanometrics (2003). *Libra VSAT Software & Hardware Manuals – Nanometrics Data Formats*.

Coordinamento editoriale e impaginazione

Centro Editoriale Nazionale | INGV

Progetto grafico e redazionale

Daniela Riposati | Laboratorio Grafica e Immagini | INGV

© 2014 INGV Istituto Nazionale di Geofisica e Vulcanologia

Via di Vigna Murata, 605

00143 Roma

Tel. +39 06518601 Fax +39 065041181

<http://www.ingv.it>



Istituto Nazionale di Geofisica e Vulcanologia