

2007

**Pubblicazione in tempo reale di
eventi sismici in formato SAC
(RT_TWF2SAC)**

Remo Moro

n.44

Istituto Nazionale di Geofisica e Vulcanologia

Via di Vigna Murata 605 - 00143 Roma

tel 06518601 • fax 065041181

www.ingv.it



**PUBBLICAZIONE IN TEMPO REALE
DI EVENTI SISMICI IN FORMATO SAC
(RT_TWF2SAC)**

Remo Moro
remo.moro@ingv.it

*Istituto Nazionale di Geofisica e Vulcanologia
Sezione CNT - Roma*

<i>Introduzione</i>	5
1. Specifiche funzionali	6
2. Configurazione delle macchine.....	7
3. Struttura degli script	8
4. Specifiche tecniche	8
4.1. <i>Ricezione della comunicazione di un evento da Locator: il modulo 'listen'</i>	8
4.2. Struttura delle informazioni: il modulo 'parser'	9
4.3. Acquisizione delle informazioni sull'evento	10
4.4. Conversione dell'informazione in SAC.....	10
4.5. Acquisizione delle informazioni su Poli&Zeri	10
5. Utilità aggiuntive	11
5.1. Esecuzione su date a scelta	11
5.2. Pubblicazione delle forme d'onda.....	11
5.3. Ricerca di eventi in un intorno della mezzanotte.....	12
5.4. Backup dei dati.....	12
<i>Conclusioni</i>	13

Introduzione

Il seguente Rapporto Tecnico spiega il funzionamento base e l'evoluzione subita dal progetto *rt_twf2sac*. La prima versione del progetto comprendeva un modulo per la ricezione di messaggi che attivasse il procedimento di conversione in tempo reale delle forme d'onda dal formato TWF [1], utilizzato all'interno dell'Istituto Nazionale di Geofisica e Vulcanologia di Roma [INGV], al formato SAC [2]. L'attuale versione aggiunge funzionalità al procedimento. I principali sono la pubblicazione su uno spazio ftp pubblico dei dati convertiti e la possibilità di poter eseguire il procedimento su dati reperiti non in tempo reale. La prima iniziale è stata ideata ed implementata da Diego Sorrentino [3]. Sulla base di tale progetto sono state aggiunte le dovute modifiche e migliorie di funzionalità esposte in questo rapporto tecnico.

1. Specifiche funzionali

I segnali della Rete Sismica Nazionale vengono elaborati da un sistema di acquisizione (Backnet[4], Locator[5]) che li memorizza in un formato proprietario TWF. Un algoritmo automatico di ricerca delle fasi sismiche (Backnet) individua segmenti di segnale significativi (detti “*trigger*”) e li memorizza in un archivio di file. Il localizzatore automatico (Locator) associa le fasi in *eventi* e calcola i parametri ipocentrali.

Il progetto *rt_twf2sac* è il sistema che si occupa di convertire in tempo reale in formato SAC binario little-endian i segnali (trigger) degli eventi localizzati; i dati in questo formato sono utilizzati per la creazione di mappe di scuotimento (Shakemaps) nell’ambito del progetto DPC-S4 [6].

Il sistema, in particolare, svolge i seguenti compiti:

1. attende comunicazione da *Locator* sul nuovo evento
2. crea la struttura ad albero sul file system che ospita le informazioni
3. reperisce le informazioni sull'evento (trigger)
4. converte le forme d’onda in SAC ed aggiunge informazioni supplementari
5. reperisce le configurazioni di Poli&Zeri [7]
6. pubblica su ftp le forme d’onda nel formato ottenuto.
7. esegue di nuovo la procedura sugli eventi del giorno precedente
8. esegue il backup dei dati su disco esterno

Dopo ogni terremoto è possibile reperire dal server ftp <ftp://iside.rm.ingv.it/events>, pubblicamente accessibile, la cartella compressa con tutte le forme d'onda associate all’evento, in formato SAC, e tutte le informazioni necessarie a ricostruire le funzioni di trasferimento delle coppie sensore/acquisitore che hanno generato le forme d’onda (Poli&Zeri).

2. Configurazione delle macchine

La situazione fisica in cui opera l'applicazione è la seguente:

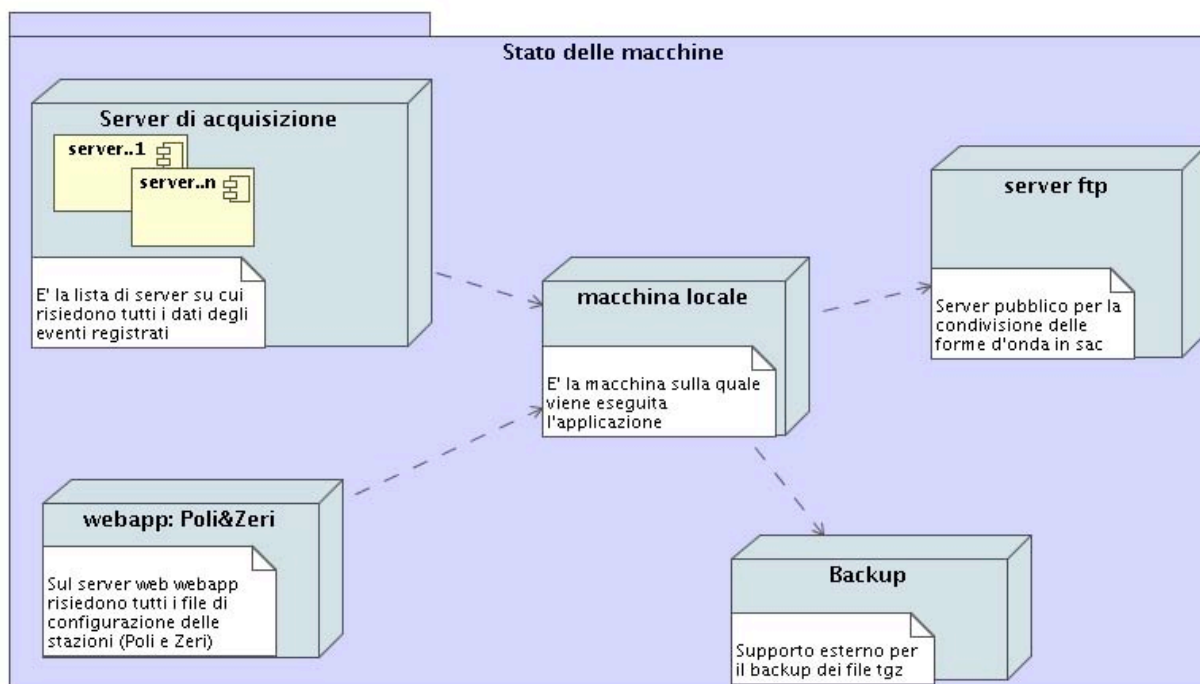


Figura 1 Schema della configurazione server-client in cui opera il procedimento di pubblicazione dei segnali sismici associati a un terremoto.

La *macchina locale* indica la macchina sulla quale è installata l'intera procedura. Questa macchina necessita di un collegamento di tipo TCP/IP (client/server) con i *server di acquisizione*, su cui operano Backnet e Locator, per reperire i dati dell'evento (segnali e parametri ipocentrali) ed interroga il server web *webapp* per ricevere i parametri strumentali e di stazione. Necessita poi di un collegamento con il *server ftp* per la pubblicazione del lavoro svolto. Per motivi di sicurezza viene anche realizzato un semplice *backup* dei dati creati giorno per giorno. Attualmente, questo sistema di backup risiede su un disco esterno montato dalla macchina locale.

3. Struttura degli script

Nel seguente diagramma vengono mostrati tutti gli script, scritti in *perl*, il programma compilato, scritto in C, che compongono l'applicazione.

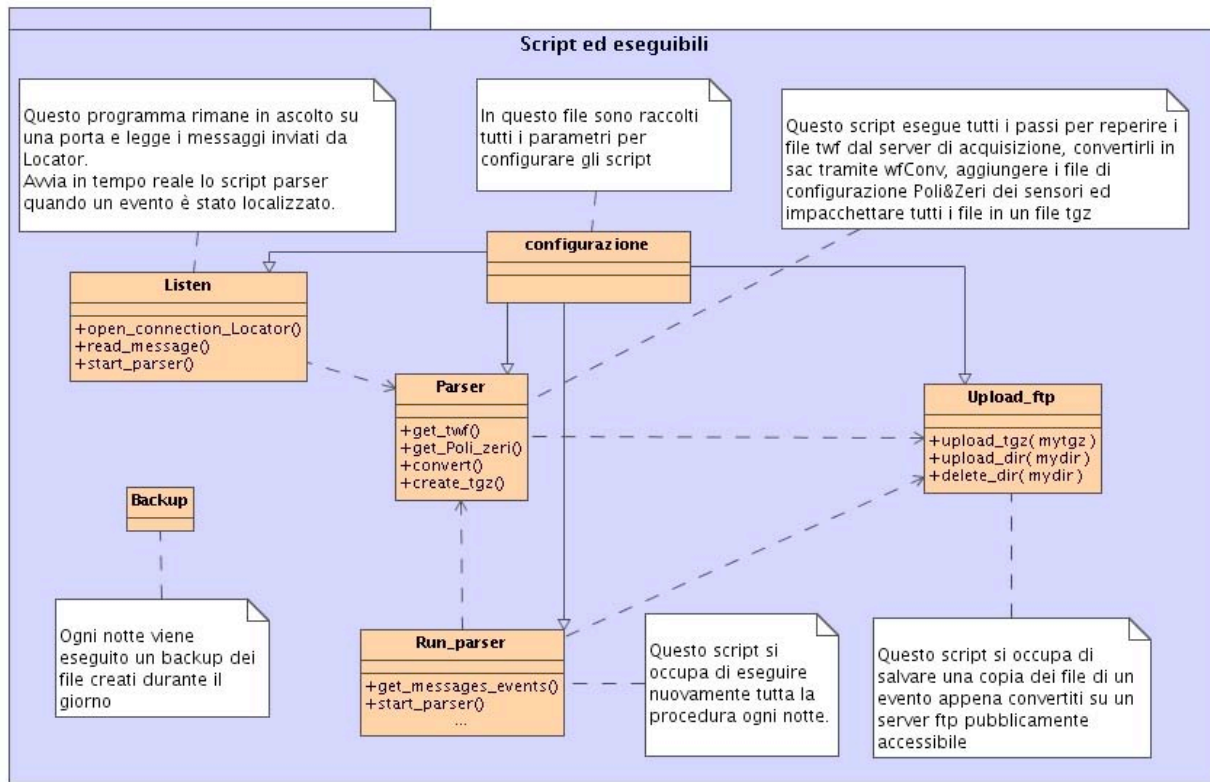


Figura 2 L'insieme dei file, script ed eseguibili, che compongono l'applicazione.

Questa figura mostra la composizione di tutti gli script e programmi che concorrono alla creazione dei file SAC. Il nome di ogni applicativo o file è indicativo delle operazioni svolte. In ogni blocco rappresentante il file o l'applicazione, presente nella figura 2, sono specificate le funzioni principali.

4. Specifiche tecniche

4.1. Ricezione della comunicazione di un evento da Locator: il modulo 'listen'

Il programma *Locator* è un processo server che invia messaggi di lunghezza predeterminata a processi client, che si collegano a un numero di porte predefinite. I messaggi contengono i parametri ipocentrali del terremoto localizzato. Il modulo *listen* è un client di *Locator* realizzato modificando l'applicazione *'read_socket'*[8]: esso ricerca su un range precedentemente definito la prima porta resa disponibile dal programma di localizzazione. È consentito definire, oltre al range di porte, anche una lista di server con i quali connettersi nel caso in cui il server a cui si è attualmente connessi sia per un motivo qualsiasi non raggiungibile o non funzionante correttamente. Attualmente, lo stato di acquisizione dei dati degli eventi è duplicato su due soli sistemi, detti *Tokyo* e *Kyoto*. I dati ricevuti sono in formato *Locator* così da non dover modificare l'applicazione esistente. L'eseguibile *listen*, scritto nel linguaggio C, riceve dal *Locator* dei messaggi inviati con una frequenza di 20 secondi. Quando non ci sono informazioni d'evento da diffondere, *Locator* genera dei messaggi di *ALIVE*, al fine di assicurare i client del proprio corretto funzionamento all'altro capo della connessione. Quando un evento sismico viene localizzato, il messaggio contiene le informazioni necessarie affinché vengano reperiti i dati acquisiti sull'evento stesso e vengano convertiti i file TWF nel formato SAC. Alla ricezione di un tale messaggio il modulo *listen* richiama un

altro processo, *parser*, scritto in *perl*, che si occupa di eseguire il resto delle procedure di seguito descritte e schematizzate nel diagramma di sequenza illustrato in figura 3.

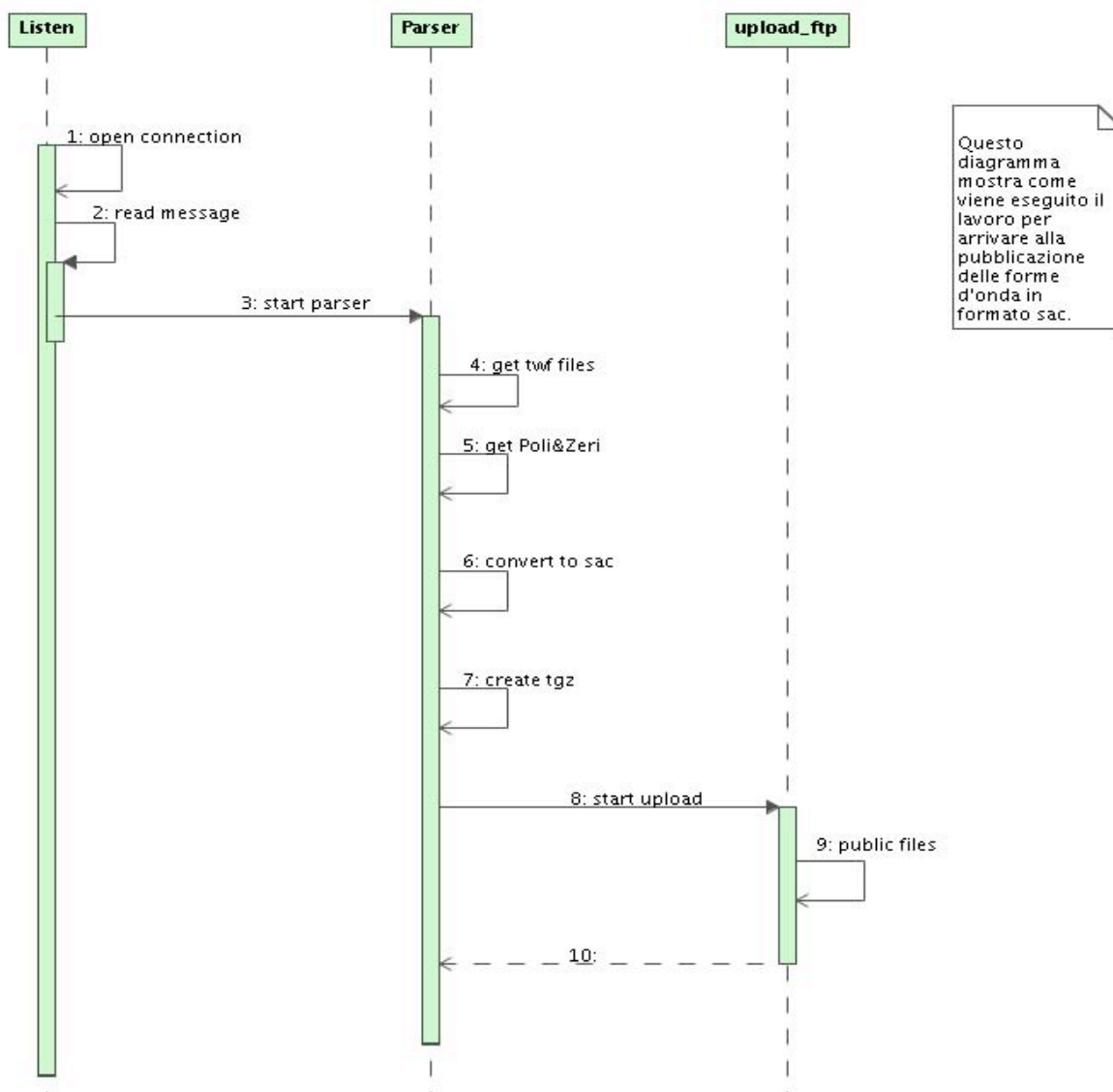


Figura 3 Diagramma di sequenza temporale dell'intero processo di pubblicazione dei segnali sismici.

4.2. *Struttura delle informazioni: il modulo 'parser'*

Una volta che il modulo *parser* viene avviato da modulo *listen* alla ricezione di un messaggio utile, esso si occupa di:

1. realizzare la struttura a directory per ospitare l'evento
2. copiare i file TWF in locale
3. reperire i file Poli&Zeri
4. avviare l'applicazione per convertire i file da TWF a SAC
5. creare il file tar.gz della directory contenente tutte le informazioni dell'evento
6. avviare la copia dei file su spazio ftp.

Il formato delle directory è il seguente: *<mount point>/YYYY/MM/DD/event_id*, dove:

- *<mount point>* è la directory/disco che ospita le informazioni
- *YYYY*, *MM*, *DD* sono rispettivamente l'anno, il mese ed il giorno in cui si è verificato l'evento

(anno in 4 cifre)

- *event_id* è una stringa di tipo *YYMMDD_HHMMX* dove *YYMMDD* ripetono l'informazione relativa al giorno dell'evento, *HHMM* sono l'ora e il minuto approssimativo in cui è accaduto l'evento ed *X* è un carattere alfanumerico aggiuntivo che serve a distinguere eventuali terremoti avvenuti nello stesso minuto.

Una volta completata la procedura, nella directory

<mount point>/YYYY/MM/DD sono disponibili i file *event_id .tar.gz* degli eventi della giornata. Ogni singolo file d'evento viene generato con un ritardo che varia dai 4 ai 7 minuti, dopo l'effettivo istante di accadimento dell'evento.

4.3. Acquisizione delle informazioni sull'evento

Dai messaggi ricevuti da *Locator* è possibile risalire al file di fasi (PHS), creato da *Locator* stesso, in cui sono esplicitati i path dei file che contengono le forme d'onda dell'evento in formato TWF. L'archivio dove trovare i file di fasi è inserito nel file di configurazione. L'applicazione reperisce il file PHS, lo analizza e copia nella struttura preposta i file indicati contenuti i segnali sismici.

4.4. Conversione dell'informazione in SAC

Una volta che i file in formato TWF sono stati copiati nella directory di lavoro, questi vengono convertiti in formato SAC. L'applicazione (SacBuilder[9]) già esistente è stata modificata allo scopo di:

- convertire file non completi [*]
- integrare le informazioni attualmente non presenti.

[*] Non è possibile assicurare che il file dell'evento sia completo al momento in cui viene copiato. Per sopperire a questa mancanza vedere la soluzione adottata nel capitolo 5.1.

4.5. Acquisizione delle informazioni su Poli&Zeri

Contestualmente alla copia delle forme d'onda dell'evento è importante che siano presenti i file di *Poli&Zeri* che indicano la configurazione della stazione e quindi degli strumenti che registrano l'evento. L'applicazione, quindi, reperisce tali file dall'applicazione *Backnet Web Control* [11] (BN_WEB) (l'applicazione BN_WEB costruisce i file di configurazione di diverse applicazioni, basandosi sulle informazioni contenute nel database SeisDB[12]). Sul server web sono presenti i file di configurazione ordinati per tempo; essi memorizzano l'intera storia delle risposte strumentali della Rete Sismica Nazionale a partire dal 2003. In questo modo è possibile conoscere la configurazione delle stazioni e degli strumenti in un momento preciso della loro vita (vedi manuale BN_WEB[11] e Sit[13]). Il modulo *parser* esegue un download dal sito BN_WEB dei file che contengono le risposte strumentali dei canali che hanno generato registrazioni da convertire in formato SAC. . Per questa operazione ci si può avvalere di prodotti pronti e perfettamente funzionanti (ad esempio `wget`[14]). Questa procedura di download dei Poli&Zeri richiede il parser della pagina fornita dal sito webapp. Attualmente è in fase di sviluppo una procedura automatica che reperisce le informazioni direttamente dal database nel quale sono contenute.

5. Utilità aggiuntive

5.1. Esecuzione su date a scelta

Finora abbiamo visto come l'intero procedimento *rt_twf2sac* viene eseguito in tempo reale al verificarsi di un evento. Tuttavia può verificarsi il caso in cui i file in formato TWF non siano ancora completamente creati al momento dell'esecuzione del *parser*, a causa spesso della grandezza dell'evento che richiede tempo per la creazione dei file stessi. La localizzazione in tempo reale definitiva, che dà inizio all'intero processo *twf2sac*, avviene prima ancora che le stazioni abbiano terminato di ricevere tutti i segnali generati dal terremoto. La procedura di creazione dei file SAC fornisce dati utili ad altre procedure che opera in tempo reale: in particolare gli applicativi che sono stati sviluppati nell'ambito del progetto DPC-S4 per il calcolo veloce del momento tensore (TDMT [15]), e per la definizione delle ShakeMaps utilizzano i dati prodotti da *twf2sac*. Per questa ragione abbiamo preferito privilegiare la velocità della realizzazione rispetto alla completezza dell'informazione. Sorge quindi la necessità di far eseguire il procedimento per convertire di nuovo i dati quando l'evento è terminato, oppure quando per qualche ragione la comunicazione tra Locator e la procedura abbia fallito. Indicando una data a propria scelta, è possibile convertire le forme d'onda di eventi avvenuti nel passato, di cui si hanno i file di evento. Per rendere automatico questo processo, viene creato un altro modulo '*run_parser*' con l'incarico di cercare da solo, utilizzando la data di riferimento, tutti i file di evento che riassumono la stessa informazione dei messaggi ricevuti in tempo reale da Locator. Il modulo *run_parser* viene eseguito ogni notte e riceve come parametri una data (una coppia di date se si vuole eseguire la procedura su uno spazio di tempo superiore ad una giornata). Quindi il modulo *run_parser* si sostituisce completamente al *listen* ed ha lo stesso scopo di avviare il modulo *parser* passandogli i dovuti parametri per la conversione del formato delle forme d'onda. In questo modo esiste la certezza che il giorno successivo i dati pubblicati saranno completi e non presenteranno delle carenze dovute a file di segnali non completi.

Come abbiamo visto il processo *rt_twf2sac* può operare su date arbitrarie; poiché l'informazione relativa alle risposte strumentali è ottenuta da *Backnet Web Control*, è stato necessario modificare questa ultima applicazione perché contenesse le risposte strumentali storiche. Può accadere che la configurazione delle stazioni, rappresentata dai file Poli&Zeri, possa differire nel tempo a causa di modifiche effettuate sugli strumenti. Per avere quindi una informazione quanto più corretta della configurazione delle stazioni in un preciso momento, abbiamo modificata l'applicazione *Backnet Web Control* in modo tale che fornisca, per ogni stazione, diversi file Poli&Zeri per le diverse configurazioni che la stazione ha assunto nel tempo. Questi file vengono creati a partire dalle informazioni presenti nel database SeisDB; in questo modo il processo *rt_twf2sac* reperisce i file Poli&Zeri corrispondenti alla data in cui si vuole calcolare la conversione delle forme d'onda. In dettaglio il modulo *parser* esegue una richiesta di una pagina al server BN_WEB, fornendo la data di riferimento ed il nome della stazione. Il server, quindi, restituisce una lista di file di configurazione della stazione, nel caso in cui la stazione abbia subito modifiche nel tempo. A questo punto il modulo *parser* può reperire il file di risposta strumentale corretto da associare alle stazioni che hanno prodotto le forme d'onda.

5.2. Pubblicazione delle forme d'onda

Nella precedente versione di *rt_twf2sac* implementata da Sorrentino, i dati in SAC di ogni evento risultavano disponibili in una directory su una singola macchina, raggiungibile solamente dall'interno della rete dell'INGV. Per rendere disponibile pubblicamente ad altri ricercatori i dati sugli eventi, si è deciso di comprimere ogni directory di riferimento all'evento in unico file tar.gz [16] e di renderlo disponibile su un server ftp, raggiungibile anche dall'esterno della rete LAN dell'INGV. Dopo aver completato la conversione delle forme d'onda di un singolo evento, la cartella che comprende tutte le informazioni sull'evento viene compressa nel formato tar.gz e lo script *upload* si occupa di depositare il

file compresso sullo spazio ftp pubblico (ftp://iside.rm.ingv.it/events/). Ogni notte viene eseguita nuovamente la procedura di conversione su tutta la giornata appena trascorsa. Lo script *upload* si occupa di cancellare tutti i file presenti nella directory del giorno in questione e di ricreare uno ad uno i file per tutti gli eventi verificatisi nella giornata scelta.

Dal giugno 2006, ad ogni evento localizzato da *Locator* viene associato un codice identificativo (id), utile nella ricerca dell'evento ed impiegato solo all'interno dell'Istituto. Per rendere quindi più semplice tale ricerca a tutti gli applicativi che utilizzano i file in formato SAC, è stato aggiunto un file *file_event_id* nella stessa cartella in cui sono presenti i file compressi di ogni giorno. In questo file ogni riga è formata da una corrispondenza uno a uno tra l'id dell'evento ed il nome del file compresso.

Di seguito un esempio:

```
2195725090:20070320_0029A.tgz
```

Il primo numero indica l'id univoco dell'evento, mentre la stringa successiva indica il nome del file compresso presente nella directory che si riferisce all'evento stesso. In questo modo conoscendo l'id univoco dell'evento, leggendo solo il file *file_event_id* è possibile recuperare tutto il file compresso delle forme d'onda senza cercare dentro ogni file compresso l'id univoco, evitando così perdita di tempo derivante dallo compattamento dei file compressi. Poiché tale id è riconosciuto solo all'interno dell'Istituto, si è pensato di non esplicitarlo nel nome del file compresso che è formato solo dalla data e dal tempo origine dell'evento.

5.3. Ricerca di eventi in un intorno della mezzanotte

Tutti i processi che gestiscono gli eventi, come *Locator*, lavorano su directory ad albero, in particolare "<anno>/<mese>/<giorno>". Il file PHS (in formato Hypoinverse 2000 [17]) che viene creato e che contiene le informazioni sulle fasi dell'evento, mostra nel suo nome l'ora della prima localizzazione dell'evento e viene salvato nel giorno in cui viene calcolata la prima localizzazione. Tuttavia, nelle successive localizzazioni, è possibile che l'ora e la data si discostino dalla prima localizzazione. In quest'ultimo caso, se ci troviamo in un intervallo di tempo attorno alla mezzanotte, capita che la cartella in cui viene salvato il file PHS non corrisponda al giorno in cui è effettivamente accaduto l'evento. Poiché per ottenere il file PHS viene eseguita una ricerca su tutti i file di un giorno utilizzando come parametro il tempo origine, è necessario eseguire una ricerca sui due giorni contigui se il tempo origine dell'evento è prossimo alla mezzanotte. Così facendo si è sicuri di trovare il file o in un giorno o in quello successivo/precedente.

5.4. Backup dei dati

Per evitare che il lavoro svolto nella conversione dei dati vada perduto al verificarsi di imprevisti, il processo di *rt_twf2sac* fornisce un semplice script per eseguire un backup incrementale. Questa operazione automatica viene eseguita ogni notte, eseguendo una copia dei dati prodotti durante il giorno, non prima però aver atteso la completa conversione dei file twf del giorno precedente.

Conclusioni

Attualmente l'intera procedura è in esecuzione su una macchina Pentium III, con 384Mb di ram. I dati prodotti dal procedimento vengono memorizzati prima sul disco locale e resi accessibili tramite protocollo *smb* sulla macchina *s4.int.ingv.it* visibile solo dall'interno della rete Lan dell'Istituto; subito dopo vengono copiati sullo spazio ftp pubblicamente accessibile al seguente indirizzo <ftp://iside.rm.ingv.it/events>.

Il modulo *listen*, come già anticipato, può utilizzare una lista di server dai quali ricevere messaggi sugli eventi localizzati. Tuttavia per rendere disgiunti e ridondanti i dati ed evitare che eventi calcolati da un sistema si confondano con quelli di un altro (Tokyo, Kyoto), vengono installati sulla macchina *s4.int.ingv.it* due processi *rt_twf2sac*, identici, ma che si mettono in ascolto ognuno su un solo server di acquisizione. Ricordo che attualmente le macchine di acquisizione nell'Istituto sono 2: *Tokyo* e *Kyoto*. In questo modo sullo spazio ftp saranno presenti due sottodirectory *tokyo* e *kyoto* nelle quali vengono memorizzati i dati acquisiti dai server omonimi. I dati convertiti in SAC sono disponibili dall'aprile 2004 per il server Tokyo, mentre per il server Kyoto solo dal novembre 2006. Per ogni anno l'occupazione di questi dati si aggira intorno ai 6GB su disco.

Riferimenti

- [1] TWF, *Temporary Wave Format*, formato delle forme d'onda dell'evento.
- [2] SAC, *Seismic Analysis Code*, Lawrence Livermore National Laboratory si veda: <http://www.llnl.gov/sac/>
- [3] Diego Sorrentino, collaboratore tecnico della sezione CNT dell'INGV di Roma, diego.sorrentino@ingv.it
- [4] Backnet, progetto curato e mantenuto da Lucio Badiali, badiali@ingv.it
- [5] Locator, progetto curato e mantenuto da Franco Mele, mele@ingv.it
- [6] Progetto DPC-S4 (2005-2007) *Stima dell'intensità dello scuotimento del terreno in tempo reale o quasi-reale per terremoti significativi ricadenti entro il territorio nazionale* (INGV-Dipartimento della Protezione Civile). Si veda: <http://earthquake.rm.ingv.it/>.
- [7] Poli&Zeri, le funzioni di trasferimento delle coppie sensore/acquisitore che generano le forme d'onda
- [8] Read_Socket, progetto curato e mantenuto da Marco Olivieri, olivieri@ingv.it
- [9] SacBuilder, progetto curato e mantenuto da Diego Sorrentino
- creato sulla base di WfConv[10]
- [10] WfConv, progetto curato e mantenuto da Lucio Badiali e Alessandro Piscini
- [11] Backnet Web Control, progetto curato da Diego Sorrentino e mantenuto da Remo Moro
- [12] SeisDB, progetto curato e mantenuto da Carlo Marcocci, marcocci@ingv.it
- [13] Manuale Sit, redatto da Diego Sorrentino e revisionato da Melissa Mendicino, melissa.mendicino@ingv.it
- [14] Wget, progetto GNU <<http://www.gnu.org/software/wget/wget.html>>
- [15] Dreger and Helmberger, 1993; Dreger, 2003: *Time Domain Moment Tensor (TDMT)*; si veda <http://seismo.berkeley.edu/~dreger/mtindex.html>
- [16] Tar, software per la compressione di file e cartelle in un unico file compresso, <http://www.gnu.org/software/tar/>
- [17] F. W. Klein, User's Guide to HYPOINVERSE-2000, U.S. Geological Survey Open File Report 02-171 V1.0