

## **STRUTTURA DEL P.N.R.A DATABASE**

(Versione 2.0, aggiornata il 20/11/03 a Baia Terra Nova)

Prof. I.Tabacco – P.N.R.A (Programma Nazionale di Ricerche in Antartide) – Dpt. Scienze della Terra, Sez. Geofisica - Università di Milano - Via Cicognara 7 - 20129 Milano  
e-mail: [ignazio.tabacco@unimi.it](mailto:ignazio.tabacco@unimi.it) tel. +39 (0) 2 50318475

Dott. A.Forieri – Dpt. Scienze della Terra, Sez. Geofisica - Università di Milano - Via Cicognara 7 - 20129 Milano  
e-mail: [alessandro.forieri@unimi.it](mailto:alessandro.forieri@unimi.it) tel. +39 (0) 2 50318471

### **INTRODUZIONE**

Il database contiene tutti i radar dati raccolti tra il 1995 e il 2003 ad opera delle differenti spedizioni del P.N.R.A.

Il database è organizzato in una struttura gerarchica costituita da **missioni**, **voli** e **tratti**. Ogni missione è costituita da uno o più voli e ogni volo è costituito da uno o più tratti.

Le missioni vengono identificate con un numero di due cifre, progressivo a partire da 01.

All'interno di ciascuna missione i voli vengono a loro volta identificati con un numero di due cifre progressivo a partire da 01.

Infine, per ciascun volo i tratti vengono identificati anch'essi con un numero di due cifre, progressivo a partire da 01.

Ogni tratto è quindi univocamente determinato da un gruppo di 6 cifre, in cui le prime due identificano la missione, le seconde due il volo della missione e le ultime due il tratto all'interno del volo. Per esempio il tratto identificato come *050204* si trova nella missione *05*, nel volo *02* ed è il numero *04*.

### **STRUTTURA DELLE DIRECTORY DI MISSIONE**

Il nome della directory è lo stesso della missione a cui essa si riferisce e cioè: Mission *XX* (*eventualmente – Nome della missione*)

Ogni directory Missione contiene:

un file Mission.XX.ps, che permette di localizzare la missione, distinguere i voli e identificare i tratti attraverso i nodi (waypoints), in una griglia di coordinate  $\varphi, \lambda$ ;

un file Mission.XX.xls, suddiviso in tre fogli (Mission ID, Flight ID e Leg ID) che sintetizza le informazioni principali relative alla missione (nome, descrizione, data, informazioni GPS, etc...), ne identifica i voli, e ne specifica i tratti;

un file Mission.XX.dat, unione di tutti i files \*.tuo (vedi dopo) della missione;

una serie di sottodirectory riferite a ciascun volo di quella missione.

### **STRUTTURA DELLE DIRECTORY DI VOLO**

Il nome della directory di volo è: Flight *XX* (*eventualmente – Nome del volo*)

La directory di volo contiene due sottodirectory, chiamate Gps e Radar Legs.

La directory Gps contiene 4 files:

file *.dat:	file contenente i dati originali GPS;
file *.ssf:	file contenente i dati GPS decodificati con il software Pathfinder Office 2.8;
file *.pos:	file contenente i dati GPS esportati in formato ASCII;
file *.wpt:	file contenente le posizioni di inizio e fine di ciascun tratto del volo.

La directory Radar legs contiene una serie di sottodirectory riferite a ciascun tratto del volo.

Per le missioni del **1995**, ovvero le missioni 01,02,03, ciascuna directory contiene 8 files:

file *.rad:	file radar originale acquisito durante la campagna antartica;
file *.bin:	file binario derivante dalla conversione del file *.rad
file *.dzt:	file derivante dalla conversione del *.bin e leggibile dal programma Radan per Windows;
file *.e.dzt:	file derivante dall'elaborazione del file *.dzt;
file *.lay:	file contenente l'identificazione dei picchi del riflettore basale e gli spessori del ghiaccio;
file *.out:	file equivalente a .lay con l'aggiunta delle coordinate geografiche;
file *.tuo:	file equivalente a .out con l'aggiunta della quota di superficie (desunta da dati ERS-1) e della quota del fondo;
file *.alt:	file contenente le quote di superficie nei punti in cui è stato fatto il rilevamento della posizione attraverso il sistema GPS.

Per missioni del **1997, 1999 e 2001**, ciascuna directory contiene 7 files o 13 files (nel caso in cui sia stato necessario spezzare in due differenti tratti un unico file \*.bin):

file *.bin:	file binario originale
file *.dzt:	file derivante dalla conversione del *.bin e leggibile dal programma Radan per Windows;
file *e.dzt:	file derivante dall'elaborazione del file *.dzt;
file *.lay:	file contenente l'identificazione dei picchi del riflettore basale e gli spessori del ghiaccio;
file *.out:	file equivalente a .lay con l'aggiunta delle coordinate geografiche;
file *.tuo:	file equivalente a .out con l'aggiunta della quota di superficie (desunta da dati ERS-1) e della quota del fondo;
file *.alt:	file contenente le quote di superficie nei punti in cui è stato fatto il rilevamento della posizione attraverso il sistema GPS.

## ATTRIBUTI DI UNA MISSIONE

### STRUTTURA DELLA TABELLA MISSION-ID (File MissionXX.xls, foglio1)

NOME CAMPO	CONTENUTO	FORMATO
Mission ID	Numero identificativo della missione	Intero 2
Mission name	Nome della missione	Testo
Description	Descrizione della missione. In particolare vengono citati il numero di voli ed il numero di tratti	Testo
Region	Zona in cui la missione si è svolta	Testo
Start Date	Data di inizio della missione	gg/mm/aa
End Date	Data di fine della missione	gg/mm/aa
Data type	Specifica il tipo di dati radar acquisiti (Airborne RES o Ground based RES)	Testo
Platform	Descrizione del mezzo dal quale sono state effettuate le misure	Testo
Thickness method	Metodo utilizzato per ottenere lo spessore del ghiaccio e frequenza operativa	Testo
Pulse length	Lunghezza dell'impulso radar	Testo
Scan samples	Numero di campioni per ciascuna traccia	Testo
Sampling interval	Intervallo di campionamento	Testo
Tracks acquisition rate	Frequenza di acquisizione delle tracce	Testo
Thickness procedure	Metodo di calcolo dello spessore del ghiaccio	Testo
Thickness logging method	Descrizione del metodo di acquisizione del dato di spessore (digitale, analogico, film 35 mm etc)	Testo
Navigation method	Metodo di navigazione ( local fix,Doppler,GPS, etc)	Testo
GPS sampling	Frequenza di acquisizione dei dati GPS	Testo
Navigation description	Descrizione dei metodi di navigazione (pseudo range, cinematico, differenziale)	Testo
Navigation precision	Precisione della misura della posizione	Testo
Terrain method	Metodo di determinazione dell'altezza dell'aereo riferita al terreno (Const. pressure, const. terrain clearance, etc.)	Testo
Terrain logging method	Descrizione del metodo di acquisizione del dato di altezza dell'aereo	Testo
Surface method	Metodo di determinazione della quota altimetrica della superficie	Testo
Surface logging method	Descrizione del dato di acquisizione della quota altimetrica della superficie	Testo
Surface precision	Precisione della misura della quota altimetrica della superficie	Testo
Reference map	Indicare se si ha una mappa di riferimento	Testo
Datum	Ellissoide di riferimento utilizzato nella determinazione della quota di superficie	Testo
Data volume Scans / Picked up scans	Numero complessivo di tracce della missione inserite nel database	Testo
Source location	Ubicazione dei dati originali	Testo
Source principle investigator	Nome della persona a cui far riferimento in caso di richiesta dei dati	Testo

## ATTRIBUTI DI UN VOLO

### STRUTTURA DELLA TABELLA FLIGHT-ID (File MissionXX.xls, foglio2)

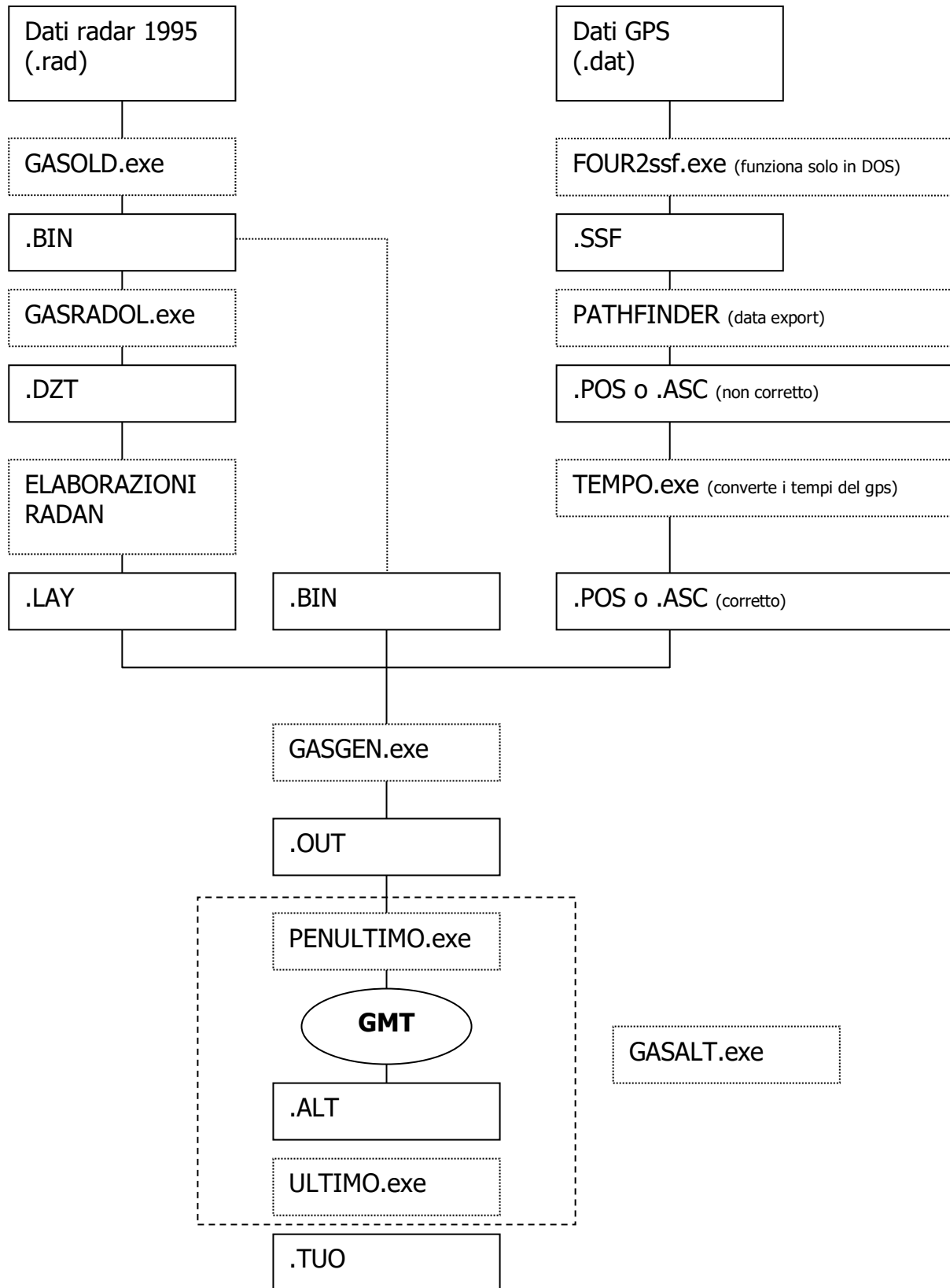
NOME CAMPO	CONTENUTO	FORMATO
mission id	numero identificativo della missione	Intero 2
flight id	numero identificativo del volo	Intero 2
flight name	nome identificativo del volo	Testo

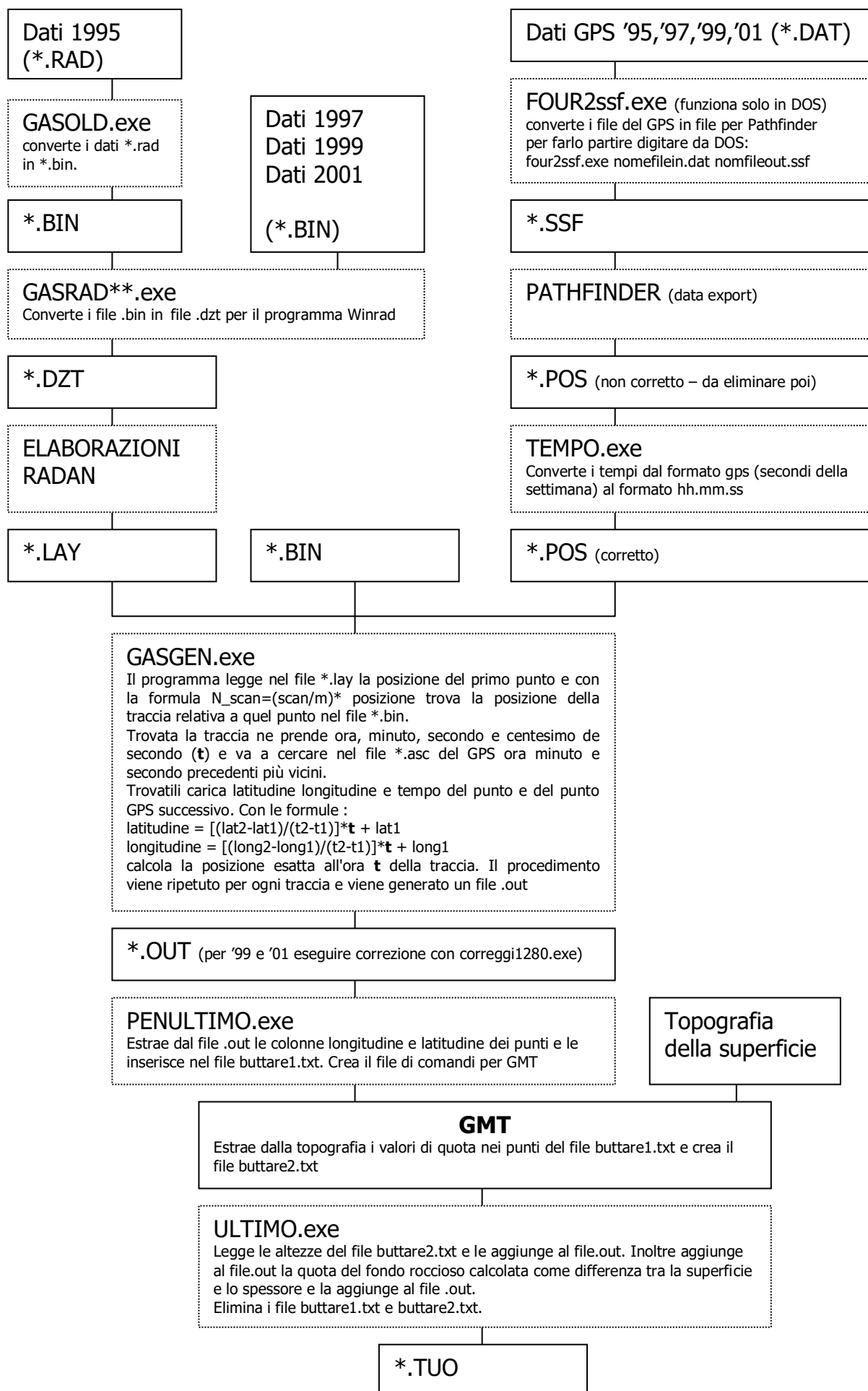
**ATTRIBUTI DI UN TRATTO**  
**STRUTTURA DELLA TABELLA LEG-ID (File MissionXX.xls, foglio3)**

NOME CAMPO	CONTENUTO	FORMATO
mission id	numero identificativo della missione	Intero 2
flight id	numero identificativo del volo	Intero 2
leg id	numero identificativo del leg	Intero 2
leg name	nome identificativo del tratto costituito dai numeri o lettere identificativi dei punti di partenza e di arrivo (es 03-06 significa leg con partenza nel punto 03 e arrivo nel punto 06)	Testo (massimo 8 car)
GPS (* .dat,* .ssf, * .pos,* .wpt)	nome dei file *.dat, *.pos, *.ssf e *.wpt. La codifica degli otto caratteri che compongono il nome per il file è la seguente: 2 lettere (ro) 2 cifre identificative dell'anno in cui i file sono stati acquisito 3 cifre identificative del giorno giuliano, cioè del giorno dell'anno in cui i file sono stati acquisiti 1 cifra numero progressivo del rov, in quanto in un giorno posso aver raccolto più file. Il numero progressivo parte da 0 Esempio: ro973452.dat è il file acquisito nell'anno 1997, giorno giuliano 345 (10 dicembre), terzo file della giornata	Testo 8 car
Radar source file (* .rad, * .bin, * .dzt)	nome del file .rad originale e dei file *.bin e *.dzt derivati (missioni compiute nel 1995) oppure nome del file *.bin originale e del *.dzt derivato (missioni compiute nel 1997, 1999 e 2001) La codifica del nome per il file *.bin dal 1997 al 2001 è la seguente: 2 lettere identificative della missione e che non hanno nulla a che vedere con il numero identificativo della stessa 4 lettere o numeri identificativi dei punti di partenza e di arrivo per il tratto 2 cifre numero progressivo del *.bin Esempio: dc030648.bin è il file binario numero 48 (ultime 2 cifre), acquisito nella missione codificata come DC da chi materialmente ha raccolto i dati e inerente il tratto che parte nel punto 03 e arriva nel punto 06 <b>NOTA:</b> nelle missioni del 1999 e del 2001 si è reso necessario spezzare alcuni file binari in due o più parti. In tal caso dopo le ultime due cifre identificative del file si sono aggiunte le lettere a,b,c,...  La codifica del nome per il file *.bin del 2003 è la seguente: 2 cifre codificati da chi ha raccolto i dati della missione e che non hanno nulla a che vedere con il numero identificativo della missione stessa 2 cifre costanti che sono 03 e identificano l'anno di escuzione delle misure 2 lettere o numeri, che insieme alle prime due cifre identificano i punti di partenza e di arrivo per il tratto (punto 0 è quello di partenza, punto 9 quello di arrivo e lettere dell'alfabeto per quelli intermedi) 2 cifre numero progressivo del *.bin Esempio: 2403bc49.bin è il file binario numero 48 (ultime 2 cifre), acquisito nella missione codificata come 24 da chi materialmente ha raccolto i dati e inerente il tratto che parte nel punto 24b e arriva nel punto 24c	Testo 7,8,9 car
radan processed file (* .dzt)	nome del file *.dzt elaborato	Testo 8 car
Ice thickness (* .lay)	nome del file *.lay Il nome del file è lo stesso del radar source file, tranne nel caso di file binari spezzati in cui si sono eliminate le due cifre identificative del file	Testo 7 o 8 car
FIR parameters	parametri di elaborazione del file *.dzt	Testo
data volo	si intende la data della partenza del volo (per non generare confusione con voli che si effettuano a cavallo della mezzanotte)	gg/mm/aa
ora di partenza	ora di inizio del tratto registrata sul file *.bin nel caso di file spezzati il punto intermedio (ovvero la fine del file denominato con a e l'inizio del file denominato con b) è stato determinato guardando sul file *.bin l'ora della n-esima traccia in cui è stato eseguito il taglio	hh.mm.ss
ora di arrivo	ora di fine del tratto registrata sul file *.bin	hh.mm.ss
coordinate partenza (long)	coordinata presa dal *.bin nel caso di file spezzati il punto intermedio (ovvero la fine del file denominato con a e l'inizio del file denominato con b) è stato determinati guardando l'ora di acquisizione della traccia in cui il file è stato spezzato e poi calcolando la sua coordinata mediante interpolazione dei file gps	Reale con 5 decimali F9.5
coordinate partenza (lat)	coordinata presa dal *.bin nel caso di file spezzati il punto intermedio (ovvero la fine del file denominato con a e l'inizio del file denominato con b) è stato determinati guardando l'ora di acquisizione della traccia in cui il file è stato spezzato e poi calcolando la sua coordinata mediante interpolazione dei file gps	Reale con 5 decimali F9.5

coordinate arrivo (long)	coordinata presa dal *.bin nel caso di file spezzati il punto intermedio (ovvero la fine del file denominato con a e l'inizio del file denominato con b) è stato determinati guardando l'ora di acquisizione della traccia in cui il file è stato spezzato e poi calcolando la sua coordinata mediante interpolazione dei file gps	Reale con 5 decimali F9.5
coordinate arrivo (lat)	coordinata presa dal *.bin nel caso di file spezzati il punto intermedio (ovvero la fine del file denominato con a e l'inizio del file denominato con b) è stato determinati guardando l'ora di acquisizione della traccia in cui il file è stato spezzato e poi calcolando la sua coordinata mediante interpolazione dei file gps	Reale con 5 decimali F9.5
N° tracce file	numero di tracce radar contenute nel file *.bin	Intero max 5
Numero di tracce analizzate	numero di tracce in cui la riflessione dal fondo è visibile e che sono state inserite nel file *.lay	Intero max 5
lunghezza tratto (km)	lunghezza del tratto calcolata con pathfinder per i file spezzati è stata ottenuta con la formula (scan)/(scan/km)	Reale con 3 decimali
scan/km	numero di tracce per km di lunghezza calcolata sempre (anche per files spezzati) come: scan totali/lunghezza totale	Reale con 3 decimali
correzione tempo (micros)	correzione eseguita al fine di identificare la riflessione dalla superficie del ghiaccio	Reale con 3 decimali
correzione quota (km)	nota la correzione tempo, essa è calcolata come correzione tempo*velocità/2	Reale con 3 decimali
note		Testo

## SCHEMI ELABORAZIONE DATI (VECCHIO – senza superficie ers-1 – E NUOVO)





# STRUTTURA DEI DIFFERENTI TIPI DI FILES DEL DATABASE E MODALITA' DI CREAZIONE

## IL FILE \*.BIN (dal 1997)

Un file binario dei dati acquisiti con il programma gas di acquisizione dati radar glaciologici composto da tre parti: intestazione, dati e coda.

### INTESTAZIONE

L'intestazione è composta da una stringa ASCII di 79+79 caratteri:

I primi 79 caratteri sono vuoti

I secondi 79 caratteri contengono l'intestazione vera e propria di cui un esempio è riportato qui sotto:

```
Ver nomefile  GGMMYY-HHmsscLAT.....Long.....Quota.      freq.      Lpulse
02.0tdb1c112.BIN211199-0541369372:43.1146S158:14.8295E2703.0  5 60  50  1 0 nonutilizzato
                                     |           |
                                     medie    n°tracce acquisite
```

4 caratteri	versione del programma utilizzata
12 caratteri	nome originale del file binario così come uscito dalla strumentazione radar
6 caratteri	giorno mese ed anno di inizio dell'acquisizione dei dati (formato ggmmyy)
1 carattere	trattino
8 caratteri	ora di inizio dell'acquisizione dei dati (formato hhmmsscc)
11 caratteri	latitudine di inizio acquisizione (presa dal gps in formato DD:mm,mmmmS)
12 caratteri	longitudine di inizio acquisizione (presa dal gps in formato DDD:mm,mmmmE)
6 caratteri	quota dell'aereo (dato assolutamente non attendibile)
7 caratteri	riservati a dati della frequenza del radar
10 caratteri	riservati a dati sulla lunghezza dell'impulso
2 caratteri	non utilizzati

### DATI ACQUISITI

Successivamente le tracce sono scritte in valori binari a 8 bit (0 - 256) in cui per ogni traccia vi sono:

quattro byte con ora, minuti, secondi e centesimi di secondo

la traccia vera e propria è, a seconda delle versioni del programma di acquisizione (campagna antartica), composta da:

ver.1.1 1024 punti spaziat di 50 ns per un totale di 51.2  $\mu$ s

ver.2.0 1280 punti spaziat di 50 ns per un totale di 64  $\mu$ s

ver.3.0 2048 punti spaziat di 50 ns (mai utilizzata)

ver.4.0 1280 punti spaziat di 50 n per un totale di 64  $\mu$ s

le tracce (ora e traccia) sono scritte in successione una dopo l'altra.

### CODA

Alla fine vi è una coda identica al "header" ma con l'ora e la posizione finale di acquisizione.

### ATTENZIONE:

Scan/s = n°tracce acquisite/medie

### NOTA BENE:

Per estendere il range del radar fino ad arrivare a 64  $\mu$ s nei dati dell'ultima campagna antartica 1999-2000 furono acquisiti 1280 byte per traccia. Queste tracce di 1280 byte nei file.bin vengono ridotti nei file.dzt a 1024 byte prendendo 4 valori su 5. Operando cioè una sorta di decimazione necessaria per il programma RADAN.

## IL FILE \*.DZT

La struttura di un file dzt è la stessa di un file \*.bin e cioè intestazione, dati e coda. Per quanto riguarda l'intestazione per il programma RADAN vi è un header di 1024 byte (lo stesse informazioni del .bin prima dette) a cui seguono le tracce ciascuna di 1024 valori (non viene più mantenuto il tempo della singola traccia). Per operare con il programma RADAN i files.bin originali vengono ridotti, se necessario a 1024 byte per traccia.

Un file .dzt, per essere letto dal programma di elaborazione Radan deve necessariamente avere tracce di 1024 (o 2048 o 4096) punti, senza l'ora in cui esse sono state acquisite.

### COME OTTENERE UN FILE \*.DZT

1. Lanciare il file gasrad\*.exe. Il programma chiede il nome del file \*.bin da convertire ed automaticamente crea un file \*.dzt con lo stesso nome del file \*.bin. Solamente per le versioni dal 1999 in avanti il programma chiede anche il nmero della traccia dalla quale iniziare la conversione.



## IL FILE \*.LAY

Un file .lay contiene 11 colonne non intestate e separate dalla virgola, contenenti i seguenti dati:

colonna	Contenuto	Formato
1	nome del file	carattere *8
2	frequenze	intero i4
3	distanza in metri dall'inizio del profilo	reale con 2 decimali f10.2
4	coordinata y del profilo	reale con 2 decimali f5.2
5	numero dello strato elaborato	intero i1
6	profondità del punto in metri	reale con 2 decimali f7.2
7	ampiezza del segnale in db	reale con 2 decimali f7.2
8	deviazione dalla curva dei minimi quadrati di tutte le riflessioni arrivate dallo stesso strato in un'area orizzontale di ampiezza pari a quella minima impostata dall'utente e centrata sul picco in esame	reale con 2 decimali f8.2
9	percentuale di scansioni che contengono la riflessione ad uno stesso livello in un'area di ampiezza pari a quella minima impostata dall'utente e centrata sul picco in esame	reale con 2 decimali f8.2
10	velocità di penetrazione del segnale nello strato	reale con 2 decimali f8.2
11	tempo di riflessione (twf) del segnale	reale con 2 decimali f7.2

## IL FILE \*.OUT

La struttura di un file.out è la stessa di un file .lay, con le colonne intestate e con aggiunte le seguenti ulteriori informazioni, riguardanti ora e posizione del punto:

colonna	Contenuto	formato
12	Ore	intero i2
13	Minuti	intero i2
14	Secondi	intero i2
15	centesimi di secondo	intero i2
16	easting	reale con 3 decimali f11.3
17	northing	reale con 3 decimali f11.3
18	Longitudine	reale con 5 decimali f9.5
19	Latitudine	reale con 5 decimali f9.5
20	altezza del ricevitore GPS	reale con 3 decimali f8.3

### COME OTTENERE UN FILE \*.OUT

Per ottenere il file .out è necessario lanciare il programma gasgen\*\*.exe

In ingresso il programma richiede:

- Il file \*.lay risultato dell'elaborazione dati con il Radan
- La lunghezza delle tracce dati nel file bin (1024 o 512)
- Il file \*.bin dei dati acquisiti con il gas.exe
- Il file \*.asc dei dati GPS elaborati con il Pfinder
- La velocità di scansione in scan/m utilizzata anche nell'elaborazione dei file dati con il Radan
- Lo strato da elaborare
- Il passo di acquisizione del GPS.

Il programma legge nel file \*.lay la posizione del primo punto e con la formula  $N\_scan = (scan/m) * posizione$  trova la posizione della traccia relativa a quel punto nel file \*.bin.

Trovata la traccia ne prende ora, minuto, secondo e centesimo de secondo (**t**) e va a cercare nel file \*.asc del GPS ora minuto e secondo precedenti più vicini.

Trovatili carica latitudine longitudine e tempo del punto e del punto GPS successivo. Con le formule :

$latitudine = [(lat2-lat1)/(t2-t1)] * t + lat1$

$longitudine = [(long2-long1)/(t2-t1)] * t + long1$

calcola la posizione esatta all'ora t della traccia. Il procedimento viene ripetuto per ogni traccia e viene generato un file .out

**ATTENZIONE:** per i file del 1999 e del 2001 si è resa necessaria la riduzione di 1 traccia da 1280 a 1024 campioni, prendendo arbitrariamente 4 campioni ogni 5 registrati. Tuttavia questo comporta un errore nell'individuazione dell'esatto tempo di riflessione in quanto nel programma RADAN non si è modificata la lunghezza complessiva della traccia che è rimasta di 64  $\mu s$ . Infatti, se una traccia lunga 64  $\mu s$  è composta da 1280 campioni, essi sono equispaziati di 50 ns, mentre se una traccia è composta di 1024 campioni, essi risultano equispaziati di 62.5 ns. Quindi, siccome il RADAN legge il tempo di riflessione e poi lo converte in spessore, il fatto di ridurre i campioni da 1280 a 1024 vuol dire modificare artificialmente il tempo di riflessione.

Per ovviare a questo inconveniente occorre lanciare il programma correggi1280.exe. Il file serve a riposizionare i tempi di una traccia a 1024 campioni nei tempi esatti di una traccia a 1280 campioni. Tale ulteriore passaggio è necessario per eliminare l'errore introdotto con il gasrad\*\*.exe.

Il programma chiede in ingresso:

- il nome del file .out da correggere
- il nome da assegnare al file .out corretto
- la correzione tempo eseguita per creare il file .lay (inserirla col segno +)

Il programma funziona nella maniera seguente:

1. lettura del tempo di riflessione e riposizionamento corretto tramite il valore della correzione tempo;
2. individuazione del campione sulla traccia a 1024 campioni.
3. riposizionamento del campione sulla traccia a 1280 campioni

calcolo del nuovo tempo di riflessione e della nuova profondità

## IL FILE \*.TUO

In un file .tuo le colonne sono separate da uno spazio e contengono, dopo una prima riga di intestazione delle colonne, i dati riportati nella tabella seguente:

colonna	Contenuto	formato
1	numero di identificazione del tratto	intero i6
2	nome del file	carattere *8
3	numero di traccia nel file dzt elaborato	intero i5
4	Frequenze	intero i4
5	distanza in metri dall'inizio del profilo	reale con 2 decimali f10.2
6	coordinata y del profilo	reale con 2 decimali f5.2
7	numero dello strato elaborato	intero i1
8	profondità del punto in metri	reale con 2 decimali f7.2
9	ampiezza del segnale in db	reale con 2 decimali f7.2
10	deviazione dalla curva dei minimi quadrati di tutte le riflessioni arrivate dallo stesso strato in un'area orizzontale di ampiezza pari a quella minima impostata dall'utente e centrata sul picco in esame	reale con 2 decimali f8.2
11	percentuale di scansioni che contengono la riflessione ad uno stesso livello in un'area di ampiezza pari a quella minima impostata dall'utente e centrata sul picco in esame	reale con 2 decimali f8.2
12	velocità di penetrazione del segnale nello strato	reale con 2 decimali f8.2
13	Tempo di riflessione (twf) del segnale	reale con 2 decimali f7.2
14	Ore	intero i2
15	Minuti	intero i2
16	Secondi	intero i2
17	centesimi di secondo	intero i2
18	Latitudine	reale con 5 decimali f9.5
19	Longitudine	reale con 5 decimali f9.5
20	Easting	reale con 3 decimali f11.3
21	northing	reale con 3 decimali f11.3
22	altezza del ricevitore GPS	reale con 3 decimali f11.3
23	Altitudine della superficie ottenuta dai dati ERS-1	reale con 3 decimali f11.3
24	Altitudine del bedrock, ottenuta sottraendo alla superficie lo spessore	reale con 3 decimali f8.3

### COME OTTENERE UN FILE \*.TUO

Lanciare il file ultimotraccia.exe. Per il corretto funzionamento deve essere presente nella directory anche il file buttare2.txt creato da job.bat

Il programma chiede in ingresso:

1. il nome del file .out
2. il numero di scan/m utilizzato per la creazione del file .lay (solo nel caso si utilizzi la versione ultimotraccia.exe)
3. il codice da assegnare al tratto

Il programma:

1. legge le altezze del file buttare2.txt e le aggiunge al file.out. creando un file .tuo.
2. ricalcola la quota dello spessore del ghiaccio moltiplicando il tempo di riflessione per la velocità dell'onda elettromagnetica nel ghiaccio
3. aggiunge al file.out la quota del fondo roccioso calcolata come differenza tra la superficie e lo spessore
4. nel caso si utilizzi la versione ultimotraccia.exe aggiunge anche il numero di traccia del file \*.dzt
5. elimina dalla directory i file buttare1.txt e buttare2.txt.

## IL FILE \*.POS

Un file \*.pos, dopo essere stato corretto, contiene 7 colonne non intestate e separate dalla virgola, contenenti i seguenti dati:

La correzione del file è necessaria in quanto ci si è accorti che il tempo gps può essere registrato dal pathfinder in due maniere differenti e non coincidenti tra loro. Infatti il pathfinder può registrare sia quello che lui chiama "time" sia quello che lui chiama "gpstime". Il primo ha formato hh:mm:ss. Il secondo è invece il numero di secondi che sono trascorsi dall'inizio della settimana. Il programma tempo converte tale tempo in giorni ore minuti secondi e lo confronta con il tempo "time". Se i due tempi coincidono non ci sono problemi, altrimenti tiene ore minuti e secondi del "gpstime".

colonna	Contenuto	Formato
1	Longitudine	Reale con 9 decimali f13.9
2	Latitudine	Reale con 9 decimali f13.9
3	Easting	Reale con 1 decimale f8.1
4	Northing	Reale con 1 decimale f9.1
5	Altgps (inutile nelle successive fasi, ma necessaria per il gasgen.exe)	Reale con 1 decimale f6.1

6	Data	GG/MM/AA
7	Ora	HH:MM:SS

ATTENZIONE: assicurarsi che nel formato data e ora tutti i numeri inferiori a 9 siano preceduti da uno 0 (es. le 5, 6 min e 12 secondi devono essere scritte 05:06:12). Assicurarsi inoltre che i dati ora siano equispaziati tra loro.

#### COME OTTENERE UN FILE \*.POS

Per ottenere un file \*.pos è necessario prima ottenere un file \*.pos non corretto e poi eventualmente procedere alla sua correzione. Per ottenere la prima versione del file eseguire le seguenti operazioni, a partire dal file \*.dat contenente i dati GPS originali

1. Lanciare, da ambiente dos, il file four2ssf.exe tramite la riga di comando four2ssf nomefilein.dat nomefileout.dat In tal modo si origina il file \*.ssf

2. Convertire il file \*.ssf in file \*.pos mediante Pathfinder. Scegliere dal menu Utility la voce Export. Seguire le istruzioni, facendo attenzione sia al corretto posizionamento dei campi (vedi tabella precedente), sia al corretto dimensionamento dei campi nel file di uscita. Inoltre assicurarsi che vengano scritti in questa prima versione del file sia il "tempo registrato" che il "tempo GPS". Quest'ultimo viene acquisito nel formato secondi progressivi a partire dall'inizio della settimana)

A questo punto il file \*.pos è stato creato. Gli orari di acquisizione dei dati dovrebbero essere equispaziati ad intervalli regolari di 5,7,10 o 15 secondi (a seconda di come è stato acquisito il file) e il valore del "tempo registrato" dovrebbe coincidere (a parte il formato) con il valore del "tempo GPS". Tuttavia questo non sempre accade e possono presentarsi tre differenti situazioni:

- i) il dato non è stato registrato;
- ii) il dato è stato registrato, ma il valore del "tempo registrato" e del "tempo GPS" non coincidono;
- iii) il dato è stato registrato, ma l'ora di acquisizione non è spaziata correttamente da quella precedente. Questo può accadere perché l'acquisizione GPS arrotonda al secondo successivo, quindi le 12.25.00,001 vengono scritte come le 12.25.01

Il file \*.pos va quindi corretto e, laddove ci fossero dei valori saltati, interpolarli opportunamente. Per fare queste passare al successivo punto 3.

3. Lanciare il file tempo5.exe. Inserire in ingresso il nome del file \*.pos da correggere, il nome desiderato per il file \*.pos d'uscita e il passo di acquisizione dei dati. Il programma crea il file \*.pos corretto, un file con le statistiche relative al numero di punti teoricamente da acquisire, realmente acquisiti ed eventualmente interpolati, un file con i soli punti interpolati ed avente estensione \*.int

#### IL FILE \*.ALT

Il file \*.alt contiene le quote della superficie nei punti rilevati dal GPS e la distanza progressiva dall'inizio del profilo. Esso è costituito da 6 colonne contenenti i dati riportati nella tabella seguente:

colonna	Contenuto	formato
1	Longitudine	reale con 9 decimali f13.9
2	Latitudine	reale con 9 decimali f13.9
3	Distanza da inizio profilo	reale con 3 decimali f10.3
4	Data del volo	gg/mm/yyyy
5	Ora del volo	hh:mm:ss
6	Quota della superficie	reale con 2 decimali f7.2

#### COME OTTENERE UN FILE \*.ALT

Lanciare il programma distanzasupe.exe.

Il programma crea il file \*.alt eseguendo le seguenti operazioni:

- 1) chiede all'utente le ore di inizio e fine del tratto della missione e la lunghezza del tratto;
- 2) calcola la durata del volo e la velocità media mantenuta dall'aereo lungo il tragitto;
- 3) cerca sul file \*.pos, gli istanti di tempo interessati dal tragitto;
- 4) crea il file \*.alt.

## APPLICATIVI E DESCRIZIONE DEL LORO FUNZIONAMENTO

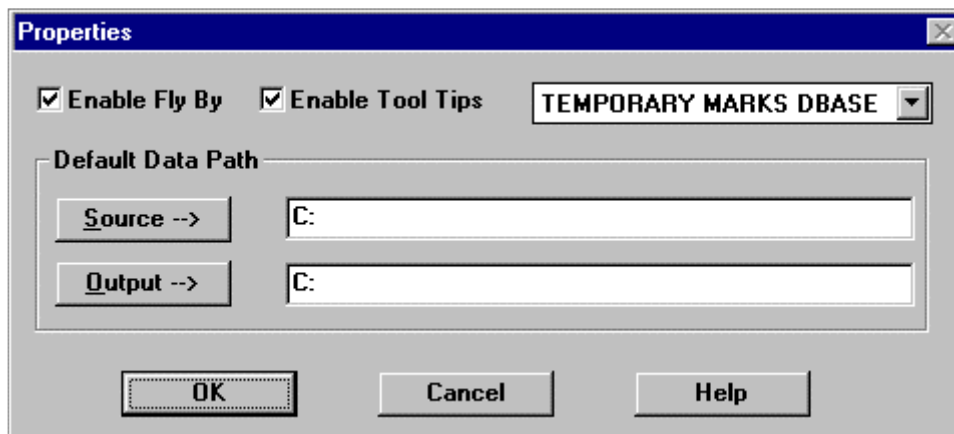
**(i codici originali e gli eseguibili si trovano in C\007\Programmi database\... prima di modificare i codici farne una copia !!! non modificare codici originali)**

<b>GASOLD.exe</b> Achille C++	converte i dati *.rad in *.bin. Il programma chiede il nome file *.rad e crea un file *.bin identico ai file generati con il gas.exe. Le tracce sono da 512 valori
<b>GASRAD**.exe</b> Achille C++	Convertire i file .bin in file .dzt per il programma Winrad  <b>NOTA PER GASRAD 99 e 01:</b> Per estendere il range del radar fino ad arrivare a 64 µs nei dati dell'ultima campagna antartica 1999-2000 furono acquisiti 1280 byte per traccia. Queste tracce di 1280 byte nei file.bin vengono ridotti nei file.dzt a 1024 byte prendendo 4 valori su 5, Operando cioè una sorta di decimazione necessaria per il programma RADAN.
<b>GASGEN**.exe</b> Achille C++	Unisce i file.bin, .lay, .pos e crea un file .out In ingresso il programma richiede: <ol style="list-style-type: none"> <li>1. Il file *.lay risultato dell'elaborazione dati con il Radan</li> <li>2. La lunghezza delle tracce dati nel file bin (1024 o 512)</li> <li>3. Il file *.bin dei dati acquisiti con il gas.exe</li> <li>4. Il file *.asc dei dati GPS elaborati con il Pfinder</li> <li>5. La velocità di scansione in scan/m utilizzata anche nell'elaborazione dei file dati con il Radan</li> <li>6. Lo strato da elaborare</li> <li>7. Il passo di acquisizione del GPS.</li> </ol> Il programma legge nel file *.lay la posizione del primo punto e con la formula $N\_scan = (scan/m) * posizione$ trova la posizione della traccia relativa a quel punto nel file *.bin. Trovata la traccia ne prende ora, minuto, secondo e centesimo de secondo ( <b>t</b> ) e va a cercare nel file *.asc del GPS ora minuto e secondo precedenti più vicini. Trovatili carica latitudine longitudine e tempo del punto e del punto GPS successivo. Con le formule : $latitudine = [(lat2-lat1)/(t2-t1)] * t + lat1$ $longitudine = [(long2-long1)/(t2-t1)] * t + long1$ calcola la posizione esatta all'ora <b>t</b> della traccia. Il procedimento viene ripetuto per ogni traccia e viene generato un file .out
<b>GASASCII.exe</b> Achille C++	Questo programma converte i file binari del WinRad *.DZT in file ASCII *.dat. Il programma chiede: <ol style="list-style-type: none"> <li>1. il nome file *.DZT</li> <li>2. la lunghezza della traccia radar (1024 o 512 byte)</li> <li>3. il numero progressivo della traccia iniziale</li> <li>4. Il numero delle tracce da convertire</li> </ol> Crea un file ASCII *.dat con le tracce disposte per colonne in una matrice di dati, leggibili da qualunque programma per la grafica.
<b>GASCONTR.exe</b> Achille C++	Questo programma serve a controllare le ore in cui sono state acquisite le tracce radar. Legge in ingresso un file .bin e scrive il numeo progressivo di traccia con l'ora di acquisizione.
<b>PENULTIMO.exe</b> Ale Fortran	Estrae dal file .out le colonne longitudine e latitudine dei punti e le inserisce nel file buttare1.txt. Crea il file di comandi per GMT Il programma chiede in ingresso: <ol style="list-style-type: none"> <li>1. il nome del file .out da aprire</li> </ol> Il programma genera: <ol style="list-style-type: none"> <li>1. il file buttare1.txt con le colonne longitudine e latitudine del file.out</li> <li>2. il file job.bat eseguibile con GMT (eventualmente cambiare il nome della griglia con la superficie)</li> </ol>

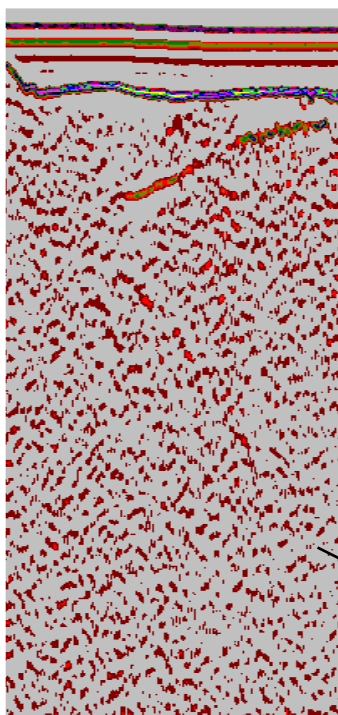
<b>CORREGGI1280.exe</b> Ale Fortran	<p>Il file serve a riposizionare i tempi di una traccia a 1024 campioni nei tempi esatti di una traccia a 1280 campioni. Tale ulteriore passaggio è necessario per eliminare l'errore introdotto con il gasrad**.exe.</p> <p>Il programma chiede in ingresso:</p> <ol style="list-style-type: none"> <li>il nome del file .out da correggere</li> <li>il nome da assegnare al file .out corretto</li> <li>la correzione tempo eseguita per creare il file .lay (inserirla col segno +)</li> </ol> <p>Il programma funziona nella maniera seguente:</p> <ol style="list-style-type: none"> <li>lettura del tempo di riflessione e riposizionamento corretto tramite il valore della correzione tempo;</li> <li>individuazione del campione sulla traccia a 1024 campioni.</li> <li>riposizionamento del campione sulla traccia a 1280 campioni</li> <li>calcolo del nuovo tempo di riflessione e della nuova profondità</li> </ol>
<b>ULTIMO.exe</b> Ale Fortran	<p>Il programma serve a creare il file .tuo completo. Per il corretto funzionamento deve essere presente nella directory anche il file buttare2.txt creato da job.bat</p> <p>Il programma chiede in ingresso:</p> <ol style="list-style-type: none"> <li>il nome del file .out</li> <li>il numero di scan/m utilizzato per la creazione del file .lay (solo nel caso si utilizzi la versione ultimotraccia.exe)</li> <li>il codice da assegnare al tratto</li> </ol> <p>Il programma:</p> <ol style="list-style-type: none"> <li>legge le altezze del file buttare2.txt e le aggiunge al file.out. creando un file .tuo.</li> <li>ricalcola la quota dello spessore del ghiaccio moltiplicando il tempo di riflessione per la velocità dell'onda elettromagnetica nel ghiaccio</li> <li>aggiunge al file.out la quota del fondo roccioso calcolata come differenza tra la superficie e lo spessore</li> <li>nel caso si utilizzi la versione ultimotraccia.exe aggiunge anche il numero di traccia del file *.dzt</li> <li>elimina dalla directory i file buttare1.txt e buttare2.txt.</li> </ol>
<b>FOUR2ssf.exe</b> File di sistema Pathfinder	<p>(funziona solo in DOS)</p> <p>converte i file del GPS in file per Pathfinder</p> <p>per farlo partire digitare da DOS:</p> <p>four2ssf.exe nomfilein.dat nomfileout.ssf</p>
<b>TEMPO.exe</b> Ale Fortran	<p>Converte i tempi dal formato gps (secondi della settimana) al formato hh.mm.ss.cs necessario per il funzionamento del gasgen.exe</p>
<b>GPSCONALT1.exe</b> Ale Fortran	<p>Il programma esegue le seguenti operazioni:</p> <ol style="list-style-type: none"> <li>legge il file *.pos</li> <li>estrae dal file *.pos le coordinate longitudine e latitudine dei punti in cui è stata rilevata la posizione dell'aereo;</li> <li>scrive tali coordinate su un file chiamato buttare1.txt. Il nome del file è stato scelto perché alla fine del processo tale file potrà essere distrutto.</li> </ol>
<b>JOB.bat</b> Ale GMT	<p>Il programma esegue le seguenti operazioni:</p> <ol style="list-style-type: none"> <li>legge il file buttare1.txt;</li> <li>legge il file con la griglia delle quote della superficie;</li> <li>per ogni punto del file buttare1.txt calcola la quota di superficie interpolando con il metodo della minima curvatura la griglia delle quote;</li> <li>crea il file buttare2.txt costituito dalle coordinate longitudine/latitudine dei punti del file buttare1.txt e dalla corrispondente quota di superficie.</li> </ol>
<b>GPSCONALT2.exe</b> Ale Fortran	<p>Il programma esegue le seguenti operazioni:</p> <ol style="list-style-type: none"> <li>legge il file buttare2.txt;</li> <li>crea un file *.pos, identico al file *.pos di partenza, in cui è aggiunta, alla fine di ciascuna riga, la quota della superficie;</li> <li>elimina i files buttare1.txt e buttare2.txt.</li> </ol>
<b>DISTANZASUPE.exe</b> Ale Fortran	<p>Il programma crea il file *.alt eseguendo le seguenti operazioni:</p> <ol style="list-style-type: none"> <li>chiede all'utente le ore di inizio e fine del tratto della missione e la lunghezza del tratto;</li> <li>calcola la durata del volo e la velocità media mantenuta dall'aereo lungo il tragitto;</li> <li>cerca sul file *.pos, gli istanti di tempo interessati dal tragitto;</li> <li>crea un file *.alt, contenente 6 colonne in cui vengono riportate: longitudine rilevata dal satellite GPS, latitudine rilevata dal satellite GPS, distanza progressiva dall'inizio del profilo, data del volo, ora del volo registrata dal GPS e quota di superficie, calcolata con GMT dall'esecuzione di job.bat</li> </ol>

## LA NORMALIZZAZIONE DELLA SUPERFICIE TOPOGRAFICA NEI FILES \*.dzt filtrati

1. lanciare l'applicativo RADAN.EXE
2. verificare che nella finestra *properties* del menu *view* i parametri siano impostati nel seguente modo:

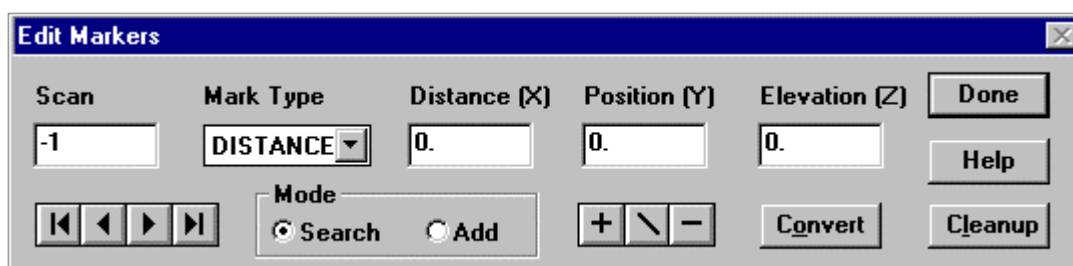


3. aprire il file da normalizzare. In questo esempio è stato utilizzato il leg 14 03 18:

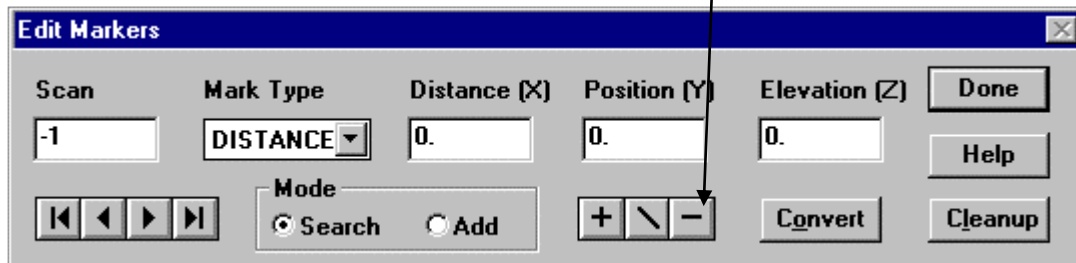


Superficie topografica da normalizzare

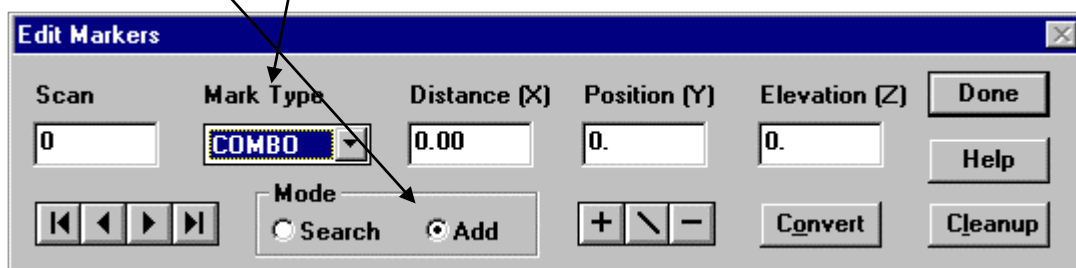
4. aprire la finestra Edit Markers (nella Toolbar, la sesta partendo da sinistra):



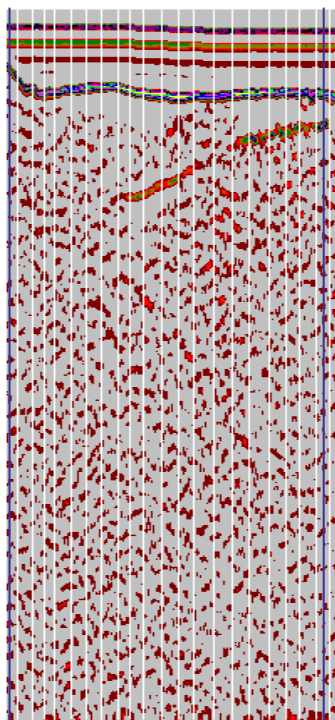
- eliminare tutti i markers eventualmente già presenti utilizzando il pulsante -:



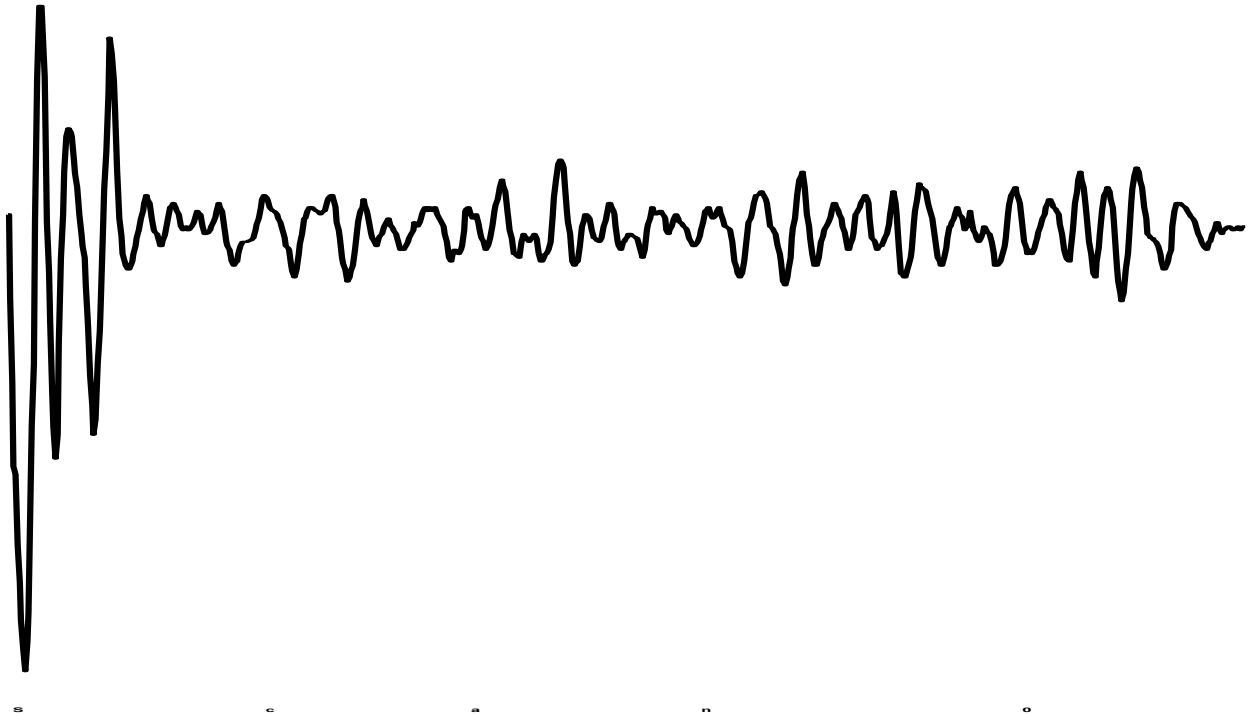
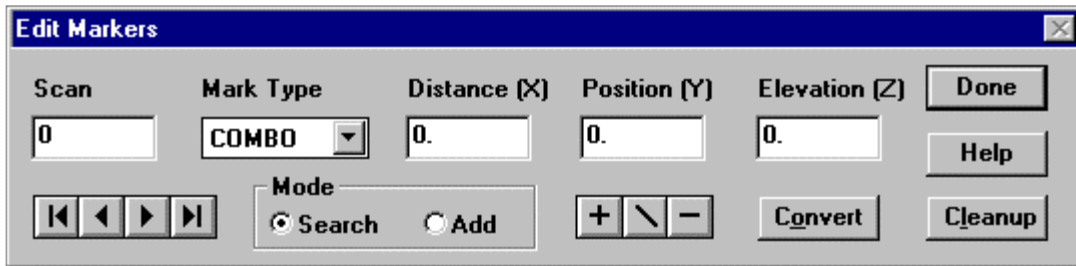
- selezionare Add e Combo nel Mark Type:



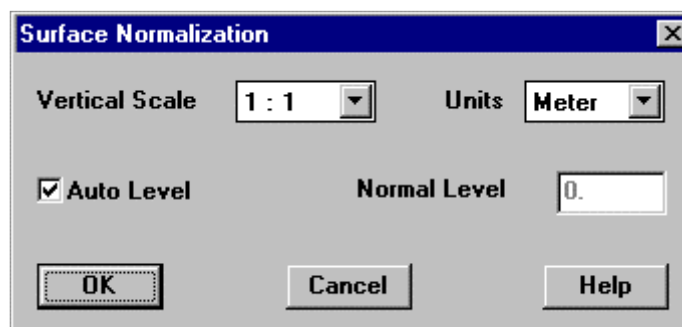
- posizionarsi con il mouse sullo scan n° 0 del file \*.dzt filtrato, tenendo il tasto sinistro ciccato; rilasciare il tasto sinistro del mouse e cliccare il + nella finestra Edit Markers . A questo punto compare il mark sul file \*.dzt filtrato. Ripetere l'operazione con una frequenza di markers variabile a seconda dell'andamento più o meno irregolare della superficie topografica (mediamente 1 mark ogni 10 scans), ricordandosi di mettere sempre un mark sull'ultimo scan del file \*.dzt. Dopo aver eseguito l'operazione, il file \*.dzt apparirà così:



8. chiudere la finestra Edit Markers con il pulsante Done. Aprire il File Header del file \*.dzt e inserire il valore della Depth in km. Riaprire la finestra Edit Markers e visualizzare il file \*.dzt in O-Scope:

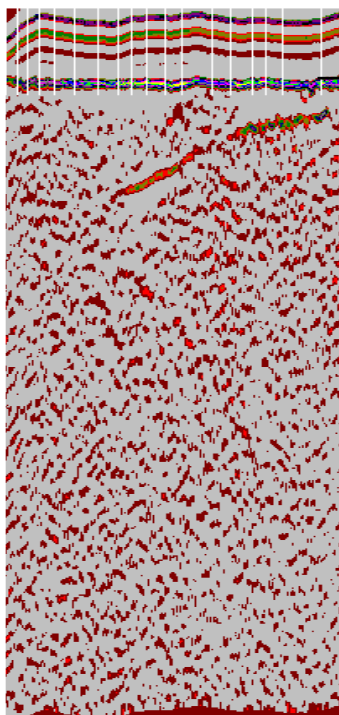


9. piccare la traccia della topografia in superficie per lo scan 0 e inserire il valore di spessore nella colonna elevation della finestra Edit Markers. Passare allo scan successivo, al quale è stato associato un mark, nella finestra Edit Markers, con il terzo pulsante in basso a sinistra; ritrovare il medesimo scan in O-Scope e inserire il valore di spessore nella colonna elevation della finestra Edit Markers. Ripetere l'operazione per tutti gli scans a cui è stato associato un mark. Alla fine chiudere la finestra Edit Markers con il pulsante Done.
10. normalizzare la topografia cliccando il terzo tasto partendo da sinistra nella Process bar (Surf Norm), dando l'OK alla finestra con i parametri impostati nel seguente modo:





11. a questo punto la superficie è normalizzata. Ricavare la *Position* e il *Top* in modo standard; inserire nell'*Header* il valore degli scan/m e il 3.2 della costante dielettrica. Si ottiene così il file *\*.dzt* elaborato:



Topografia normalizzata