

NUMERO 1 / 1981

A. FICARRA e M. NANNI

RAPPORTO VAX 11/780

CONSIGLIO NAZIONALE DELLE RICERCHE

ISTITUTO DI RADIOASTRONOMIA

c/o ISTITUTO DI FISICA « A. RIGHI »
Via Irnerio, 46 - 40126 BOLOGNA (Italy)

INDICE

INTRODUZIONE.....	1
ORGANIZZAZIONE DEL SERVIZIO.....	4
GENERALITA' SULL'ARCHITETTURA VAX (PARTE I).....	14
DUE NUOVI COMANDI PER POSIZIONARE IL NASTRO MAGNETICO.....	17
LIBRERIE DI SUBROUTINES.....	20
DA_NASTRO: UN COMANDO PER COPIARE FILES DA NASTRO A DISCO...	26
SU_NASTRO: UN COMANDO PER COPIARE FILES DA DISCO A NASTRO...	32
LIBERA_NASTRO: UN COMANDO PER CANCELLARE FILES DA NASTRI...	35
USO DEL BATCH E DELLE CODE DI STAMPA.....	39
NUOVI COMANDI DI VARIA APPLICAZIONE.....	45
PRECESSIONI ASTRONOMICHE.....	48
LIBRERIA DI PROGRAMMI PER ESPERIMENTI VLBI.....	53
STARLINK: UN ESEMPIO DELL'USO DEL VAX IN ASTRONOMIA.....	55
PROSSIMAMENTE.....	58
DOCUMENTAZIONE.....	60

INTRODUZIONE

Nasce, con questo primo numero, il "Rapporto" sull'attività di gestione dell'elaboratore VAX 11/780 della Digital.

Il "Rapporto", senza volersi in alcun modo sostituire ai manuali d'uso del VAX, la cui consultazione resta comunque indispensabile, si presenta sotto un duplice aspetto: quello di un manuale divulgativo che incorassi i non-ancora-utenti a servirsi di questo prestigioso elaboratore, immesso sul mercato da soli due anni e sia universalmente riconosciuto come il numero uno nel campo dei mini; e, insieme, quello, rivolto agli attuali utenti, di un bollettino puntuale dell'attività svolta dal servizio, con l'informazione sulla esistenza e le istruzioni per l'uso dei nuovi comandi, programmi, procedure e librerie che progressivamente verranno messi a disposizione per l'uso generale.

L'accesso a questo rapporto è aperto a tutti: chiunque, o perché a conoscenza di nuove tecniche o programmi messi a punto in qualche altra parte del mondo, o perché autore esli stesso di lavori nel campo della elaborazione dati che possano presentare un carattere di utilità generale, potrà chiedere, anzi in genere sarà vivamente pregato, di usare il "Rapporto" per divulgare alla comunità degli utenti le informazioni in suo possesso.

Nel contempo, ogni critica e ogni suggerimento che potranno rendere il "Rapporto" più funzionale ai suoi scopi saranno bene ac-

cetti e tenuti nella dovuta considerazione.

Lo scopo di chi gestisce il centro e' di giungere gradualmente all'impiego dell'elaboratore nel pieno delle sue possibilita' e cioe', non solo come un puro esecutore di calcoli, ma anche come un sistema che acquisisca dati dagli strumenti, governi processi in tempo-reale, permetta l'elaborazione interattiva di immagini a colori ecc... La tendenza sara' quella di potenziare l'elaboratore con tutto il software che e' possibile trasferire da altri centri, privilegiando ovviamente le applicazioni che riguardano il campo dell'astronomia ottica e radio, ma anche cercando di dotare il centro di tutto cio' che puo' essere utile alla comunita' scientifica in generale; contemporaneamente il "Rapporto" dovra' costituire il veicolo di scambio delle idee e di trasmissione delle informazioni senza le quali lo stesso potenziamento dell'elaboratore servirebbe indubbiamente a poco.

Lo stile, il tono e la sostanza delle cose dette in questo primo numero del "Rapporto" sono ancora in una fase sperimentale, come del resto in una fase sperimentale e' ancora tutta l'organizzazione del centro; piu' avanti, attraverso l'approfondimento della conoscenza del sistema, l'acquisizione di una maggiore esperienza di problemi gestionali e, fatto non trascurabile, la prevista assunzione di nuovo personale in organico, si potra' avviare quell'ambizioso progetto di potenziamento dell'elaboratore di cui si e' gia' parlato.

Conseguentemente anche il tono del "Rapporto" si adattera' al livello generale; per ora il lettore dovra' accontentarsi, anche se alcuni degli argomenti trattati gli sembreranno banali o di scarsa

utilita' e altri viceversa gli sembreranno incomprensibili.

Per concludere, il "Rapporto" uscirà per ora senza una periodicità prefissata. Questo per due motivi, in un certo senso opposti: anzitutto, a causa dell'estrema carenza di personale attualmente addetto al Vax, non si può essere in grado di garantire precise scadenze nel caso che la periodicità sia fissata a intervalli di tempo troppo stretti; dall'altra parte, fissando una periodicità a intervalli larghi, si corre il rischio di fornire certe importanti notizie con considerevole ritardo. Si preferisce quindi non fissare delle scadenze e fare uscire il rapporto ogniqualvolta lo si reputerà necessario, in base all'entità e alla importanza delle nuove informazioni da comunicare agli utenti.

ORGANIZZAZIONE DEL SERVIZIO E INFORMAZIONI TECNICO-BUROCRATICHE

PERSONALE ADDETTO

L'Istituto di Radioastronomia, che ha dal C.N.R. il compito di gestire il servizio del Vax, ha messo a disposizione un'unita' di personale laureato (A.Ficarra) con compiti insieme di responsabile del centro e di addetto al software, e un tecnico (ancora da assumere), con compiti di operatore. Inoltre ha richiesto al C.N.R. l'assunzione di nuovo personale in organico, esclusivamente dedicato al Vax, che possa garantire un livello minimale di gestione degli impianti e di assistenza al servizio e sia adeguato all'attuale configurazione del sistema e agli sviluppi previsti dell'immediato futuro.

Tale personale dovrebbe consistere in :

- 1) Due "collaboratori tecnici professionali" con il compito di gestire il sistema operativo. Queste persone dovrebbero anche occuparsi di programmi applicativi riguardanti problemi scientifici, soprattutto nel campo dell'astronomia ottica e radio.
- 2) Due "assistenti tecnici professionali" con la funzione di operatori. Sarà loro compito assicurare la continuita' del servizio nell'arco delle ore diurne e provvedere al rifornimento del materiale di consumo necessario.
- 3) Un amministrativo con la mansione di gestire i conti relativi agli utenti e di occuparsi degli ordini e delle fatturazioni.

L'esperienza ci dice che queste richieste, per quanto, ripetiamo, minimali rispetto alle esigenze che si presenteranno in un prossimo

futuro, non saranno completamente soddisfatte. Comunque, anche nel migliore dei casi, per il protrarsi del blocco delle assunzioni a livello di tutto il C.N.R., per molti mesi ancora il responsabile del centro sarà anche l'unica persona ufficialmente addetta al servizio.

ACCESSO ALL'ELABORATORE E TARIFFAZIONE

Nessuna formalità è per ora richiesta a chi voglia diventare utente del Vax. In questa prima fase, che possiamo definire di espansione, non si pongono limiti a priori all'uso dell'elaboratore; chiunque lavori nell'ambito C.N.R. o universitario può, dietro semplice richiesta verbale al responsabile del centro, ottenere un codice di autorizzazione e una quota di spazio-disco disponibile. Eventuali limitazioni e selezioni verranno fatte in seguito, se il ritmo di potenziamento del sistema non riuscirà più a mantenere il passo con la crescita dell'utenza.

Per quanto riguarda i costi di utilizzo, il C.N.R. ha stabilito espressamente che entro il più breve tempo possibile devono essere fissate le tariffe d'uso dell'elaboratore, differenziate secondo le singole voci (tempo-macchina, spazio-disco, materiale di consumo ecc...), e devono essere inviate regolari fatture ai codici di account a cui fanno capo i diversi gruppi di utenti. Infatti il sistema è stato acquistato sui fondi del capitolo centralizzato per l'informatica (gli stessi a cui, per esempio, attinge il CNUCE di Pisa) e si configura come nodo specializzato della rete CNUCE, cui sarà anche materialmente collegato fra pochi mesi. Quindi non è di uso esclusivo di un singolo istituto, ma deve porsi come servi-

zio a pagamento di erorazione del calcolo per il C.N.R. o altri enti affini.

Le tariffe saranno fissate seguendo gli stessi criteri adottati dal CNUCE (un decimo del costo reale delle risorse di calcolo, prezzo di costo per il materiale di consumo ecc...); tuttavia, in attesa che si costituisca quella struttura di personale in grado di assolvere alle mansioni burocratico-amministrative che competono alla sezione dei codici di acconto, e' inevitabile che in questa prima fase l'uso del calcolatore sia gratuito.

ATTIVITA' DEL SERVIZIO

L'elaboratore e' in genere disponibile per gli utenti tutti i giorni lavorativi, salvo una mezza giornata alla settimana, indicativamente il martedi' mattina, durante la quale viene fatta la manutenzione ordinaria del sistema. Interruzioni straordinarie sono inoltre possibili per esigenze di servizio, dovute soprattutto alla necessita', abbastanza frequente in questa prima fase, di operare dei tests sul sistema o procedere ad aggiornamenti e inserimenti di nuove "utilities".

L'orario giornaliero di apertura e chiusura non e' ancora fissato in modo preciso: dipende strettamente dalla presenza fisica dell'unico addetto al sistema o di qualche suo collega autorizzato e in grado di eseguire la procedura di accensione o di spegnimento della macchina. Indicativamente si puo' dire che l'elaboratore e' disponibile all'incirca dalle 8,30 alle 19,00. Non si prevede, nemmeno a tempi lunghi, di poter estendere il servizio anche alle ore notturne, salvo casi specialissimi di comprovata e

impellente necessita'.

ATTUALE CONFIGURAZIONE E DISLOCAZIONE DELLE PERIFERICHE

L'elaboratore VAX 11/780 e' stato acquistato dal Consiglio Nazionale delle Ricerche sui fondi del capitolo centralizzato per la informatica e assegnato in gestione all'Istituto di Radioastronomia di Bologna. Per delibera della Presidenza del C.N.R. a seguito del parere della Commissione per l'Informatica, il centro di calcolo cosi' costituito fara' parte della rete del CNUCE come "nodo specializzato all'elaborazione interattiva dell'immagine astronomica", intendendosi puntualizzare, con questa definizione restrittiva, quella che in prospettiva sara' la principale attivita' del centro: infatti il sistema sara' collegato entro breve tempo a una serie di terminali video a colori attrezzati per l'elaborazione interattiva delle immagini, cui l'elaboratore centrale fornira' il supporto necessario di potenza di calcolo e di memoria di massa. A sua volta il Vax, per l'esecuzione di lavori di tipo "batch" di particolare complessita', fungera' da terminale degli IBM 370/158 e 370/168 del CNUCE.

Ovviamente questa prevista specializzazione del sistema non escludera' l'uso dell'elaboratore per altri scopi: come e' sia' stato detto altrove, il centro ha l'obbligo istituzionale di fornire un servizio di calcolo, in primo luogo per gli utenti dell'area C.N.R. e secondariamente per quelli di altri enti (Universita', INFN ecc...) che svolgono ricerche nel campo delle discipline scientifiche.

L'elaboratore, nella sua configurazione base, e' stato acqui-

stato sui fondi del bilancio del 1979 e installato il 27/10/80.

Si compone dei seguenti elementi (tutti forniti dalla Digital):

- a) Elaboratore centrale, con memoria MOS di 512K byte, "Virtual Memory Management", "Floating Point Accelerator", console microprogrammata con "floppy-disk", "Multiplexer" per 8 linee asincrone (colloquio EIA) e sistema operativo comprendente il "Virtual Memory System" e il compilatore "Fortran IV Plus".
- b) Unità disco RM03 con capacità di 67M byte, contenente il sistema operativo e i files degli utenti.
- c) Unità nastro magnetico TE16 a 9 tracce, con densità 800/1600 bpi e velocità 45 ips.
- d) Telescrivente LA120, con velocità di stampa di 180 cps.
- e) 5 video alfanumerici VT100, con velocità di trasmissione fino a 9600 baud.

A queste unità bisogna anche aggiungere un video grafico Tektronix 4025, acquistato precedentemente e collegato come terminale remoto del CNUCE; attualmente, sull'uscita del terminale, è montato un commutatore che permette, a scelta, il collegamento al Vax o all'IBM.

Il Vax è stato installato in un locale del seminterrato dell'Istituto di Fisica. Nella stessa stanza in cui risiede l'elaboratore centrale si trovano anche l'unità disco, l'unità nastro, la telescrivente (usata in generale come stampante) e un video VT100 (collegato come terminale di console). Sempre nel seminterrato, in una stanza attigua, si trovano un altro video e l'impianto sia predisposto per accogliere il primo video a colori che sarà collegato all'elaboratore. I rimanenti terminali sono stati collocati negli uffici dell'Istituto di Radioastronomia, al

terzo piano dell'Istituto di Fisica. Il collegamento, realizzato dal personale dell'Istituto, e' stato fatto acquistando e stendendo cinque cavi di tipo telefonico a sette fili ed ha dato risultati eccellenti in quanto, nonostante il basso prezzo dei cavi (rispetto alle tariffe della Digital), la loro lunghezza (cento metri) e la velocita' di trasmissione mantenuta alta (9600 baud), non ha evidenziato finora nessun tipo di errore di trasmissione. Tre dei cinque cavi suddetti si collegano ai rimanenti video VT100, collocati, uno nell'ufficio del responsabile, uno nella stanza dei terminali IBM e l'ultimo in una stanza del lato ovest dell'edificio. Dei rimanenti due cavi, uno e' connesso tramite commutatore al video Tektronix e, infine, uno e' libero per collegamenti futuri di strumentazione di laboratorio.

La seguente tabella riassume la situazione delle periferiche, indicando nell'ordine il nome fisico della porta di uscita dallo elaboratore (le porte da TTA0 a TTA7 fanno parte del multiplexer a 8 linee asincrone tipo DZ11 con colloquio EIA), il tipo di unita' collegata, la funzione svolta e la sua dislocazione nello Istituto di Fisica.

NOME	UNITA'	FUNZIONE	DISLOCAZIONE
CSA1	Floppy-disk	Floppy di console	Seminterrato, stanza principale
DRA0	Disco RM03	Disco sistema	Seminterrato, stanza principale
MTA0	Nastro TE16	Nastro Magnetico	Seminterrato, stanza principale
OPA0	Telescrive LA120	Stampante (Spool Device)	Seminterrato, stanza principale
TTA0	Video VT100	Terminale di Console	Seminterrato, stanza principale
TTA1	Video VT100	Terminale inter- attivo	Seminterrato, stanza di appoggio
TTA2	Video VT100	Terminale inter- attivo	Piano 3, ufficio del responsabile
TTA3	-----	Acquisizione dati (uso futuro)	Piano 3, stanza della strumentazione
TTA4	Video VT100	Terminale inter- attivo	Piano 3, stanza dei terminali IBM
TTA5	Video TEK 4025	Terminale inter- attivo(part-time)	Piano 3, stanza dei terminali IBM
TTA6	Video VT100	Terminale inter- attivo	Piano 3, stanza lato ovest
TTA7	-----	Uso futuro	Non collegato

SVILUPPI PREVISTI NELL'IMMEDIATO FUTURO

Sui fondi del capitolo centralizzato, assegnati per il 1980, il C.N.R. ha gia' emesso ordine di acquisto delle seguenti unita':

- 1) Espansione del multiplexer DZ11, con acquisizione di 8 nuove uscite seriali asincrone (da TTB0 a TTB7), a cui sara' possibile collegare altrettanti terminali interattivi.

- 2) Interfaccia DR11-B, parallela a 16 bit. Servira' per il collegamento con il video a colori.
- 3) Interfaccia DUF11-DA seriale asincrona, con velocita' fino a 9600 baud, da usarsi per il collegamento remoto con l'IBM del CNUCE.
- 4) Interfaccia KG11-A per il protocollo BSC (in accoppiata con la precedente, sempre per il collegamento con l'IBM).
- 5) Lettore di schede perforate CR11-A con velocita' di 300 schede/minuto.
- 6) Stampante LP11S-YD, con 132 posizioni e velocita' di 600 linee/minuto.
- 7) Packase Software per l'emulazione dei terminali IBM 2780/3780. Operera' sotto il controllo del VMS del Vax, da cui sara' visto come uno dei tanti processi contemporaneamente in esecuzione e permettera' al Vax di funzionare insieme come elaboratore autonomo e come terminale "batch" dell'elaboratore IBM 370 del CNUCE.
- 8) Display Generator, modello Aydin 5218 della Technitron (video a colori), previsto nella seguente configurazione:
 - a) 8 moduli di memoria di 1024 x 1024 punti, che permetteranno 256 livelli diversi di intensita' (o 256 diversi colori).
 - b) Packase Firmware per grafica a due e tre dimensioni, con generatore di vettori e cerchi, zoom, tavole per la definizione di pseudo-colori ecc...
 - c) Joystick, con l'acquisizione dell'input grafico su comando manuale.
 - d) Tastiera alfa-numerica per i comandi, provvista in particolare di 10 tasti-funzione, tasti per il controllo del colo-

- re ecc...
e) Monitor da 19 pollici.
9) Plotter elettrostatico, modello Versatec V-80 della Technitron, con risoluzione di 200 punti per pollice e 32 livelli di grigio, provvisto di interfaccia per il collegamento al Vax e di software integrato per il sistema operativo VAX/VMS.
10) Stampante bidirezionale DP-300 della Technitron, con 132 posizioni e velocità di 300 linee/minuto.

L'installazione delle unità sopraelencate è prevista in un periodo di tempo che va dal prossimo aprile alla fine dell'anno, salvo complicazioni. Inoltre, per quanto riguarda il processo delle immagini, saranno necessari alcuni mesi prima di acquisire le competenze necessarie alla gestione e all'uso dell'unità video Audin 5218, che, come si intuisce dalla descrizione, non è un semplice terminale interattivo, ma piuttosto un elaboratore autonomo specializzato a svolgere particolari funzioni.

Quanto finora descritto fa parte della configurazione del sistema quale era stata decisa fin dall'inizio e che, solo per motivi di bilancio, è stata suddivisa dal C.N.R. in due parti: la prima, inserita nel bilancio del '79 e installata da pochi mesi, la seconda, assegnata all'80 e ordinata alle ditte fornitrici il 31 dicembre (!), con il solito ritardo fisiologico di circa un anno.

Tuttavia, un esame più approfondito di detta configurazione, confermato dall'esperienza di questi primi mesi di lavoro sul sistema, ha reso evidente un certo sottodimensionamento della memoria, centrale e di massa, rispetto al carico dell'utenza.

Per rimediare a questo squilibrio e' stato inserito nel preventivo dell'81 l'acquisto delle seguenti unita':

- 1) 1M byte di memoria centrale
- 2) 1 unita' disco RM03 da 67M byte
- 3) 1 unita' nastro TE16

Così' configurato, cioè con 1,5M byte di memoria, 2 unita' disco e 2 unita' nastro, il sistema sarà in grado di fornire un servizio efficiente al livello delle sue possibilità dichiarate e si prevede che non avrà bisogno di ulteriori ampliamenti per diversi anni.

GENERALITA' SULL'ARCHITETTURA VAX/VMS (PARTE I)

(Comincia, con questo primo numero, una serie di brevi articoli divulgativi sulle principali caratteristiche del sistema VAX/VMS, articoli rivolti soprattutto ai non-utenti, e scritti con l'intendimento di illustrare i vantaggi offerti da questo sistema, che si presenta molto potente e allo stesso momento versatile e facile da usare.)

La sigla VAX sta per Virtual Addressing eXtension ed e' stata scelta dalla casa costruttrice, la Digital Equipment Corporation, per evidenziare quella che e' la principale caratteristica del suo nuovo elaboratore, caratteristica che lo differenzia nettamente dalla serie precedente dei PDP-11 : la possibilita' di lavorare con memorie virtuali.

In modo del tutto trasparente all'utente che manda in esecuzione un programma, il sistema si incarica di suddividere il programma, inizialmente residente su disco, in porzioni di 512 byte, dette "pagine" e di trasferire nella memoria fisica solo le pagine che servono di volta in volta. Se nuove pagine sono richiamate in memoria e la porzione di memoria fisica assegnata all'utente ("working-set") e' piena, allora il sistema, o rioccupa lo spazio delle pagine non ancora modificate (e quindi di nuovo reperibili dal file originario su disco), o libera comunque lo spazio necessario trasferendo alcune pagine modificate in un file di sistema ("paging-file"), da cui le pagine sono in ogni caso sempre recu-

Perabili per un nuovo eventuale richiamo in memoria.

Questa operazione di trasferimento avanti e indietro fra la memoria e il disco si chiama "Paging" ed e' eseguita dal sistema automaticamente; l'utente puo' costruire programmi con dimensioni praticamente illimitate, avendo l'illusione di lavorare sempre in memoria fisica; l'unico problema consiste nel fatto che un eccessivo paging appesantisce il lavoro del sistema e incide sul tempo di esecuzione del programma. La frequenza del paging, tuttavia, dipende non solo dalle dimensioni del programma in se', ma anche dalla sua struttura; e' evidente che programmi che procedono sequenzialmente, cioe' senza troppi salti da un punto all'altro, saranno piu' efficienti e subiranno meno paging di programmi strutturati in modo disordinato. A questo proposito esiste addirittura una nuova branca dell'informatica, la "Programmazione Strutturata", che insegna a programmare secondo regole precise, proprio allo scopo di ottimizzare l'efficienza dell'esecuzione, evitando per esempio salti fra un settore e l'altro del programma, analizzando i dati nella stessa sequenza in cui gli stessi sono immagazzinati in memoria ecc...

Un'altra caratteristica particolare del Vax, che sta a monte e rende possibile la stessa organizzazione a memorie virtuali, e' l'indirizzamento a 32 bit, in luogo dei 16 bit standard per i PDP e la maggiorparte degli altri mini. Con la parola a 32 bit, ogni locazione di memoria virtuale puo' essere indirizzata direttamente fino a un limite teorico di 2^{32} , cioe' di circa 4 miliardi di locazioni. In pratica il limite di estensione della memoria virtuale e' sempre fissato dalla capacita' del disco, o dei dischi, che forniscono il supporto per il "paging-file" e per le

altre aree di sistema coinvolte nel "memory-management".

Nonostante queste sostanziali innovazioni che lo differenziano nettamente dagli elaboratori delle generazioni precedenti, il Vax resta del tutto compatibile con i calcolatori della serie PDP-11, nel senso che il sistema operativo dei PDP si ritrova come sottoinsieme del sistema operativo del Vax; ne consegue che qualunque programma o dato scritto per un PDP e' accessibile dal Vax (ovviamente non e' vero il contrario); per di piu' il passaggio da un sistema all'altro non e' eseguito a generazione, ma e' istantaneo, permettendo cosi' a piu' utenti di lavorare contemporaneamente nei due sistemi diversi. Infatti un registro della CPU, la "Processor Status Longword" (PSL), che indica in ogni istante lo stato del sistema, contiene un bit che segnala se in quell'istante il sistema sta lavorando in modo "nativo" (cioe' come Vax) o in modo "compatibile" (cioe' come PDP) e questo bit viene di volta in volta aggiornato dal sistema in funzione della particolare operazione che l'elaboratore sta eseguendo.

E' chiaro che la Digital, attenendosi a questo criterio generale di compatibilita' nella scelta della linea evolutiva dei suoi elaboratori, ha realizzato un importante obiettivo, anche dal punto di vista commerciale: il vecchio cliente, come il nuovo, e' invogliato ad acquistare il Vax a causa delle radicali innovazioni che vi ritrova, senza pero' la riserva mentale che avrebbe se dovesse riconvertire tutto il lavoro prodotto in precedenza.

DUE NUOVI COMANDI PER POSIZIONARE IL NASTRO MAGNETICO

Il linguaggio di cui l'utente dispone per colloquiare con l'elaboratore, il "Digital Command Language" o DCL, comprende un esteso insieme di comandi per eseguire le funzioni di:

- 1) Sviluppo interattivo di programmi.
- 2) Gestione di periferiche e di files.
- 3) Esecuzione e controllo di programmi interattivi e "batch".

Fra tali comandi non sono ancora compresi quelli per posizionare un nastro magnetico. Operazioni di input-output di questo tipo possono essere eseguite soltanto a programma tramite richiamo a particolari subroutines di sistema ("System Services"), la cui funzione è appunto quella di gestire gli scambi con le periferiche. Per ovviare a questa mancanza sono stati scritti due programmi, uno per riavvolgere un nastro e l'altro per saltare, avanti e indietro, un certo numero di files. Ciascun programma è mandato in esecuzione da un comando, esattamente come se facesse parte del DCL.

REWIND

Il comando REWIND (forme abbreviate: REWIN, REWI, REW), produce il riavvolgimento di un nastro magnetico; ovviamente il nastro deve essere già montato (comando MOUNT) e on-line.

Il comando è applicabile a qualunque tipo di nastro, standard o "foreign": infatti il programma lanciato dal comando REWIND consulta una tabella di sistema per scoprire se il nastro è standard o no e, a seconda del risultato, chiama la "System Service" appro-

priata (nel caso di nastro standard la subroutine non solo produce materialmente il riavvolgimento, ma aggiorna anche una tabella di sistema con l'informazione che il nastro si trova posizionato all'inizio).

Sintassi:

REWIND dev

Operandi:

dev : nome fisico dell'unita', valido anche nella forma generica.

MTA0:,MTA:,MT0:,MT: sono tutti operandi validi (anche senza il segno :). Notare che nella situazione attuale non c'e' ambiguita', essendo il centro dotato di una sola unita' nastro.

Se si omette l'operando, questo viene esplicitamente richiesto (appare la "prompt-string": UNIT ?)

Esempio:

REW MTA

Risultato: il nastro montato sull'unita' MTA0: e' posizionato all'inizio.

SKIP

Il comando SKIP (forma abbreviata: SKI) causa lo spostamento di posizione del nastro magnetico di un certo numero di files, avanti o indietro, a seconda del numero fornito come operando. In ogni caso il nastro si posiziona all'inizio del file raggiunto dopo l'operazione. Se appare la scritta BOT CONDITION significa che il nastro, procedendo all'indietro, ha raggiunto la posizione iniziale ("Besinnins Of Tape").

Poiche' non e' prevista l'operazione di "skip-file" per i nastri standard, nemmeno a livello di "System Service", il comando non puo' aggiornare le tabelle di sistema; ne consegue che un normale comando DCL riguardante il nastro magnetico, fornito dopo la esecuzione del comando SKIP, potrebbe dare risultati imprevedibili; per cui si consiglia di usare il comando SKIP solo per nastri "foreign" o montati come tali.

Sintassi:

SKIP dev,nf (la virgola fra i due operandi e' obbligatoria)

Operandi:

dev : come per il comando REWIND

nf : numero di files da "skippare":

se $nf > 0$ il nastro procede in avanti (Skip Forward)

se $nf = 0$ il comando non ha effetto

se $nf < 0$ il nastro procede all'indietro (Skip Backward)

Se gli operandi sono entrambi omessi, il programma li richiede esplicitamente (appare la "prompt-string" :

UNIT,#FILES (>0:F,<0:B) ?)

Esempi:

1) SKI MT,2 (supponiamo che il nastro sia inizialmente posizionato sul file N.3)

Risultato : il nastro e' posizionato all'inizio del file N.5

2) SKI MT,-3 (nastro inizialmente sul file N.8)

Risultato : il nastro e' all'inizio del file N.5

3) SKI MT,-4 (nastro inizialmente sul file N.2)

Risultato : appare la scritta: BOT CONDITION

LIBRERIE DI SUBROUTINES

E' stato creato un nuovo "directory" di sistema, di nome [LIBR], destinato a contenere tutte le librerie di programmi e subroutines di utilita' generale che saranno in futuro scritti e messi a disposizione degli utenti.

Attualmente l'unico file contenuto nel directory [LIBR] e' la libreria IRA1OBJ, costituita da alcune subroutines che eseguono operazioni di lettura, trasferimento e conversione di dati provenienti da altri elaboratori (soprattutto IBM).

La libreria IRA1OBJ si compone delle seguenti subroutines:

FROMTOA

INTX2

INTX4

READFOR

REALX4

WRITEEOF

WRITEFOR

La libreria e' richiamata automaticamente dal linkase-editor. Non e' quindi necessario che l'utente la riferisca in modo esplicito nel comando LINK.

La parte che segue descrive la funzione e l'utilizzo di ognuna delle subroutines della libreria IRA1OBJ.

FROMTOA

La subroutine FROMTOA converte dati dal codice EBCDIC (usato dall'IBM) al codice ASCII (usato dal Vax). Serve per trasferire

sui supporti del Vax dati alfanumerici scritti col "format" da un elaboratore IBM 370.

Ordine di chiamata:

CALL FROMTOA(E,A,N)

dove :

E e' il vettore contenente i dati in codice EBCDIC

A e' il vettore che dovra' contenere i dati convertiti in codice ASCII (puo' anche coincidere con E)

N e' il numero di caratteri (byte) da convertire.

INTX2

La subroutine INTX2 trasforma numeri interi di 16 bit (INTEGER*2) dal formato IBM al formato Vax. Per ogni numero la subroutine inverte di posizione il primo byte col secondo. Infatti nell'organizzazione IBM i numeri interi sono memorizzati a partire dal byte che contiene le cifre piu' significative, mentre per il Vax e' il contrario.

Ordine di chiamata:

CALL INTX2(IB,IV,N)

dove :

IB e' il vettore contenente i dati interi nel formato IBM

IV e' il vettore che dovra' contenere gli stessi dati nel formato Vax (puo' anche coincidere con IB)

N e' il numero di byte (cioe' il numero di dati x 2) da convertire.

INTX4

La subroutine INTX4 svolge la stessa funzione della subroutine INTX2, salvo il fatto che converte numeri interi a 32 bit (INTEGER*4).

Ordine di chiamata:

CALL INTX2(IV,IB,N)

dove :

IB e' il vettore contenente i dati interi nel formato IBM

IV e' il vettore che dovra' contenere gli stessi dati nel formato Vax (puo' anche coincidere con IB)

N e' il numero di byte (cioe' il numero di dati x 4) da convertire.

READFOR

La subroutine READFOR legge un blocco fisico di dati da supporto non standard. E' stata scritta soprattutto allo scopo di leggere i nastri magnetici di tipo "foreign", in particolare di provenienza IBM.

Ordine di chiamata:

CALL READFOR(ICH,L,IBU,*n)

dove :

IBU e' il vettore che dovra' contenere i dati letti

L e' il numero di byte che si prevede di leggere e, di ritorno dalla subroutine, il numero di byte effettivamente letti

ICH e' il numero di canale associato. Per ogni operazione di input-output il sistema deve conoscere un numero da associare all'operazione stessa, a cui poi fara' riferimento per

dove :

- RI e' il vettore contenente i dati reali nel formato IBM
- RV e' il vettore che dovra' contenere gli stessi dati nel formato Vax (puo' anche coincidere con RI)
- N e' il numero di byte (cioe' il numero di dati x 4) da convertire

WRITEEOF

La subroutine WRITEEOF scrive un end-of-file su nastro. E' da usarsi esclusivamente con nastri "foreign"; infatti, per i nastri standard, il sistema si incarica automaticamente di costruire sia le labels di testa che quelle di coda (con i relativi end-of-file) di ogni file che viene scritto su nastro.

Ordine di chiamata :

CALL WRITEEOF(ICH)

dove :

ICH e' il numero di canale associato (vedi READFOR).

WRITEFOR

La subroutine WRITEFOR scrive un blocco fisico di dati su nastro. E' da usarsi esclusivamente con nastri "foreign".

Note:

- 1) Poiche' la minima lunghezza di un blocco in scrittura su nastro e' di 14 byte, la subroutine si incarica, nel caso di records piu' corti, di riempire la parte rimanente con "blanks".
- 2) E' indispensabile che, terminata la scrittura del file, sia eseguita anche la subroutine WRITEEOF, meglio ancora se chiamata

due volte (doppio end-of-file, a indicazione della fine fisica dei dati su nastro).

Ordine di chiamata:

CALL WRITEFOR(BUF,L,ICH)

dove :

BUF e' il vettore dei dati da scrivere su nastro

L e' il numero dei byte da scrivere

ICH e' il numero di canale associato (vedi READFOR).

DA_NASTRO : UN COMANDO PER COPIARE FILES DA NASTRO A DISCO

Il comando DA_NASTRO manda in esecuzione una procedura interattiva che permette all'utente di trasferire su disco, un file alla volta, tutto o parte del contenuto di un nastro.

La lettura del nastro e' effettuata per blocchi fisici, cioe' con trasferimento immediato di ogni blocco in memoria, senza l'intervento del sottosistema "RMS" (la sigla sta per "Record Management Services"). L'RMS, chiamato in causa da istruzioni ad alto livello, come la READ o la WRITE del fortran o i comandi DCL, si occupa di interpretare, a seconda del tipo di struttura del file dichiarato nella "header" del file stesso, il contenuto dei blocchi fisici letti e di creare, riconoscendo le parole di controllo all'interno dei blocchi, i diversi records logici da trasferire in memoria. In questo modo vengono letti, o scritti, i files di tipo standard del Vax. Risulta chiaro quindi che un trasferimento alla cieca di un blocco fisico in memoria ha senso solo se i dati di provenienza sono non standard (in un solo caso le due operazioni sono equivalenti: quando i records logici sono di lunghezza fissa e coincidente con quella del blocco fisico).

Si consiglia perciò di usare il comando DA_NASTRO solo per copiare files non standard, come per esempio quelli costruiti da altri elaboratori, come l'IBM; il che del resto e' lo scopo principale per cui la procedura e' stata scritta. Per i nastri standard la normale operazione di copiatura e' eseguita dal comando COPY del DCL.

DESCRIZIONE DELLA PROCEDURA

La procedura chiede all'operatore di montare il nastro e si ferma in attesa della risposta; ricevuta la risposta dalla console dell'operatore, esegue il "mount" logico del nastro, qualificandolo come "foreign" e chiede all'utente di specificare il numero d'ordine del file su cui posizionarsi; se l'utente batte uno zero, la procedura esegue il riavvolgimento del nastro e poi di nuovo chiede il numero del file. Dopo avere posizionato il nastro all'inizio del file specificato, la procedura chiede all'utente il nome che il file dovrà avere quando sarà trasferito su disco. Ricevuta risposta, chiede se il nastro ha records "fissi-bloccati" di tipo IBM; una risposta positiva a questa domanda significa che l'utente intende "sbloccare" i records, cioè dividere il blocco fisico letto su nastro in tanti records da trasferire separatamente su disco; in questo caso la procedura chiede anche quale dovrà essere la lunghezza dei records; se invece la risposta è negativa si intende che ogni record letto dovrà essere trasferito così com'è, qualunque sia la sua lunghezza. In ogni caso il file di destinazione su disco è sempre definito di lunghezza variabile ed è costruito in modo standard (cioè gestibile con normali comandi DCL o leggibile a programma da istruzioni fortran tipo READ).

Infine la procedura, prima di iniziare la copiatura vera e propria, chiede all'utente se vuole la conversione EBCDIC-ASCII; una risposta positiva significa che il file di input contiene records soltanto con caratteri alfanumerici di tipo IBM e che l'utente vuole che tali caratteri siano convertiti nel formato ac-

cessibile dal Vax.

Terminata la copiatura del file senza errori (in caso contrario si ha una fine forzata), la procedura si appresta a copiare un nuovo file e ritorna a chiedere all'utente qual'è il numero del file su cui posizionarsi, ripercorrendo poi la sequenza di operazioni già viste. Questo "loop" è interrotto quando l'utente, copiati tutti i files che intendeva copiare, dà una risposta nulla (cioè batte soltanto "return") alla nuova richiesta del numero del file; allora la procedura riavvolge e scarica il nastro e poi esce (la stessa operazione finale viene eseguita nel caso di fine forzata per errore).

La procedura usa, per il posizionamento del nastro, i comandi REWIND e SKIP. Ogni volta mantiene il ricordo del numero dell'ultimo file copiato, a cui somma 1 (in quanto, alla fine della copiatura, il nastro si trova posizionato all'inizio del file seguente); sottrae il risultato dal numero del nuovo file, ottenendo così un numero che, fornito come parametro al comando SKIP, indica di quanti files si deve spostare il nastro per posizionarsi sul file desiderato. Nel caso questo file preceda come posizione su nastro l'ultimo file copiato, sta all'utente scegliere, a seconda della distanza fra i due files, se gli conviene raggiungere il nuovo file procedendo all'indietro o riavvolgere prima il nastro e poi procedere in avanti; nel primo caso deve fornire immediatamente il numero del file, nel secondo deve prima rispondere zero, provocando così l'esecuzione del comando REWIND, e poi fornire il numero del file.

Nel caso di copiatura da nastri con "labels", come per esempio gli "standard-labels" IBM o gli standard Vax, il numero del

file deve essere fornito tenendo conto anche della presenza delle labels di testa e di coda, che sono contate dalla procedura come dei normali files, in quanto il nastro e' montato "foreign". Detto S il numero del file su nastro standard e detto F il numero che si ottiene contando anche le labels, la relazione e' :

$$F = S * 3 - 1.$$

Per quel che riguarda la operazione di copiatura vera e propria, la procedura manda in esecuzione un programma, scritto in fortran, che a sua volta chiama le subroutines READFOR (per la lettura su nastro) e FROMETOA (per l'eventuale conversione EBCDIC-ASCII).

La procedura ha una malfunzione, peraltro inevitabile, da cui ci sembra corretto mettere in guardia gli utenti: se i records su nastro hanno lunghezza variabile, gli eventuali records piu' corti di 14 byte (scritti da altri calcolatori che non hanno questa limitazione, come per esempio i CDC) non sono riconosciuti dal sistema e vengono saltati, cioe' non sono copiati su disco : l'utente rischia di ritrovarsi un programma fortran senza certe istruzioni, tipo STOP, END o altre istruzioni brevi ! Purtroppo questa limitazione e' al livello hardware e non c'e' per il momento niente da fare.

USO OPERATIVO DEL COMANDO DA_NASTRO

L'utente deve fornire, insieme al comando DA_NASTRO, anche il nome del nastro da cui intende copiare i files. Questo nome ver-
ra' inserito dalla procedura nel messaggio che richiede all'ope-
ratore di montare il nastro: non serve quindi che coincida con

l'eventuale "label" di volume registrata su nastro; l'unica cosa importante e' che sia lo stesso nome scritto sull'etichetta esterna. Dopodiche', montato il nastro, la procedura si svolge in modo puramente interattivo; l'utente deve solo attendere le richieste del calcolatore e rispondere appropriatamente.

L'esempio che segue illustra un possibile uso del comando DA_NASTRO; per maggiore chiarezza i messaggi del calcolatore sono in lettere maiuscole, quelli dell'utente in lettere minuscole (i numeri sono sempre scritti dall'utente).

da_nastro mytape

(appare sulla console il messaggio:

PER FAVORE MONTA IL NASTRO MYTAPE, GRAZIE

la procedura si ferma in attesa che l'operatore fornisca la risposta)

FILE N. (0 = REWIND) : 3

NOME DEL FILE DI OUTPUT SU DISCO : myfile.dat

IL NASTRO HA RECORDS FISSI BLOCCATI ? si

LRECL ? 80

CONVERSIONE EBCDIC-ASCII ? no

FILE N. (0 = REWIND) : 8

NOME DEL FILE DI OUTPUT SU DISCO : program.for

IL NASTRO HA RECORDS FISSI-BLOCCATI ? no

CONVERSIONE EBCDIC-ASCII ? si

FILE N. (0 = REWIND) : 0

FILE N. (0 = REWIND) : 1

NOME DEL FILE DI OUTPUT SU DISCO : last.dat

IL NASTRO HA RECORDS FISSI-BLOCCATI ? no

CONVERSIONE EBCDC-ASCII ? no

FILE N. (0 = REWIND) : <return>

A questo punto il controllo ritorna al DCL. L'utente ritrova nel proprio directory tre nuovi files : MYFYLE.DAT, PROGRAM.FOR e LAST.DAT, che sono rispettivamente la copia dei files 3, 8 e 1 del nastro MYTAPE. In particolare:
LAST.DAT e' una copia esatta
PROGRAM.FOR e' stato convertito da EBCDC a ASCII
in MYFILE.DAT ciascun record originario e' stato suddiviso in records lunghi 80 byte.

SU_NASTRO : UN COMANDO PER COPIARE FILES DA DISCO A NASTRO

Il comando SU_NASTRO manda in esecuzione una procedura interattiva che permette all'utente di trasferire files da disco a nastro. L'utente puo' stabilire da quale posizione del nastro deve partire la copiatura (salvando cioe' un certo numero di files preesistenti); da quel punto in poi i files vengono trasferiti, a uno a uno, nell'ordine in cui l'utente li sceglie.

Ogni record logico letto da disco (con l'istruzione fortran READ) viene fisicamente scritto su nastro, senza nessuna trasformazione, "saltando" cioe' la gestione dell'RMS (per una breve descrizione dell'RMS vedi: DA_NASTRO). Il nastro e' quindi accessibile solo come "foreign", anche perche' la procedura si limita a copiare file per file senza aggiungere "labels" di qualche genere o altra informazione di sistema. Per copiare files da disco a nastro standard invece, si deve usare il comando COPY del DCL. Del resto la procedura invocata dal comando SU_NASTRO e' stata scritta per altri scopi, che sono soprattutto due:

- 1) Trasferire, tramite nastro magnetico, dati dal Vax a altri elaboratori.
- 2) Fare copie nastro-nastro usando il disco come supporto intermedio. In questo caso il lavoro procede attraverso due steps: prima, con il comando DA_NASTRO, si trasferiscono tutti i files dal primo nastro su disco, senza "sbloccature" e senza conversioni di codice; successivamente, con il comando SU_NASTRO, si riscrivono gli stessi files sul secondo nastro. Notare che finche' il centro dispone di una sola unita' nastro questa resta l'unica procedura possibile.

DESCRIZIONE DELLA PROCEDURA

La prima parte e' identica alla procedura del comando DA_NASTRO: viene inviato un messaggio all'operatore con la richiesta di montare il nastro. Ottenuto il nastro, montato come "foreign", la procedura chiede all'utente il numero d'ordine dell'ultimo file del nastro da non cancellare, cioe' il numero del file dopo cui posizionarsi per iniziare la scrittura. Se l'utente risponde zero significa che il nastro deve essere posizionato all'inizio, cioe', o e' nuovo, o comunque deve essere riutilizzato interamente. Dopodiche' la procedura chiede il nome del primo file da copiare ed esegue la copia, alla fine della quale scrive un end-of-file su nastro. Chiede allora il nome di un altro file che scrive su nastro di seguito al precedente e cosi' via, finche' l'utente non batte un semplice "return" (risposta nulla). A questo punto la procedura scrive due end-of-file consecutivi (end-of-tape) ed esce dopo aver riavvolto e scaricato il nastro.

La procedura usa il comando SKIP per il posizionamento iniziale del nastro. Anche qui come nel comando DA_NASTRO e' compito dell'utente calcolare l'esatto numero di files da saltare nel caso che il nastro abbia le "labels", contando quest'ultime come dei normali files (vedi la formula nel capitolo DA_NASTRO).

Per quello che riguarda la copiatura vera e propria la procedura manda in esecuzione un programma. Questo a sua volta chiama le subroutines WRITEFOR e WRITEEOF, che eseguono la scrittura rispettivamente dei dati e degli end-of-file.

USO OPERATIVO DEL COMANDO SU-NASTRO

L'utente deve fornire, insieme al comando SU_NASTRO, anche il nome del nastro su cui intende copiare i files. Successivamente dovrà scrivere, dietro richiesta della procedura, il nome dei files nello stesso ordine in cui intende che gli stessi siano copiati su nastro.

L'esempio che segue illustra un possibile uso del comando SU_NASTRO; la convenzione è la solita: lettere maiuscole per i messaggi del calcolatore, lettere minuscole per le risposte dell'utente.

su_nastro ontape

(appare sulla console il messaggio:

PER FAVORE MONTA IL NASTRO ONTAPE, GRAZIE

la procedura si ferma in attesa che l'operatore fornisca la risposta)

FILE DOPO CUI POSIZIONARSI (0 = INIZIO NASTRO) : 6

NOME DEL FILE DI INPUT SU DISCO : first.lis

NOME DEL FILE DI INPUT SU DISCO : second.dat

NOME DEL FILE DI INPUT SU DISCO : <return>

A questo punto il controllo ritorna al DCL. L'utente ritrova sul nastro ONTAPE, a partire dal file 7, la copia, nell'ordine, dei files: FIRST.LIS e SECOND.DAT.

LIBERA_NASTRO: UN COMANDO PER CANCELLARE FILES DA NASTRI STANDARD

E' noto che la differenza principale fra un disco e un nastro consiste nel fatto che ogni file su disco e' accessibile direttamente e quindi puo' essere creato, modificato o cancellato indipendentemente dalla presenza degli altri files, mentre i files su nastro sono disposti l'uno di seguito all'altro. Di conseguenza:

- 1) ogni nuovo file deve essere scritto in coda a quelli preesistenti
- 2) soltanto l'ultimo file puo' essere modificato
- 3) cancellare un file significa in realta' cancellare anche tutti i files successivi; in particolare la cancellazione si effettua semplicemente scrivendo una marca di end-of-tape al posto del file in questione.

Nel caso di nastri standard, a queste limitazioni, peraltro ovvie e inevitabili in quanto connesse con la natura stessa del supporto magnetico, se ne aggiungono altre, dovute invece alle regole imposte dal software di gestione dei nastri. La struttura dei files standard prevede che ogni file sia accessibile tramite il nome, come se risiedesse su disco: l'utente e' dispensato dall'occuparsi del posizionamento del nastro, di cui invece si incarica il sistema, facendo riferimento a certe tabelle che aggiorna a ogni nuova operazione. Come contropartita a questa indubbia comodita', il sistema impone le seguenti limitazioni:

- 1) non esistono comandi per il posizionamento del nastro (che sarebbero invece molto utili, soprattutto per nastri "foreign"); questo problema e' gia' stato discusso e risolto nei capitoli

precedenti

2) e' inibita la cancellazione dei files.

Il comando LIBERA_NASTRO rimedia a quest'ultima limitazione, mandando in esecuzione una procedura che cancella un file indicato dall'utente (e, ovviamente, tutti i files successivi).

Per le ragioni esposte finora e' evidente che il comando LIBERA_NASTRO e' da usarsi esclusivamente con nastri standard.

DESCRIZIONE DELLA PROCEDURA

La parte iniziale e' identica a quella dei comandi DA_NASTRO e SU_NASTRO: la procedura richiede all'operatore di montare il nastro e si ferma in attesa della risposta. Montato il nastro (qualificato "standard" e quindi con controllo che la label coincida col nome fornito dall'utente), la procedura chiede all'utente il nome del file da cancellare. Dopodiche' esamina il "directory" completo del nastro. Se il file indicato dall'utente non esiste, la procedura esce con un messaggio di errore; altrimenti ricava il numero d'ordine del file e scrive su terminale il nome di tutti i files che seguono quello indicato. L'utente puo' cosi' controllare l'elenco dei files che saranno cancellati ed e' ancora in tempo a evitare l'operazione, nel caso si accorga di qualche errore. Infatti la procedura chiede conferma: se la risposta non e' SI, esce senza fare nulla; se invece la risposta e' SI, calcola il numero di files che deve saltare per posizionarsi sulla label di testa del primo file da cancellare e, in quel punto, scrive un doppio end-of-file a indicazione della fine dei dati ('end-of-tape').

Alla fine la procedura riavvolse e scarica il nastro.

USO OPERATIVO DEL COMANDO LIBERA_NASTRO

E' del tutto identico a quello sia' visto per i comandi DA_NASTRO e SU_NASTRO. L'unica differenza consiste nel fatto che, trattandosi di nastro standard, il nome del nastro, fornito dall'utente insieme al comando, e' usato sia nel messaggio all'operatore che nel comando MOUNT e quindi deve anche coincidere con il nome con cui il nastro e' stato inizializzato (registrato nella "label di volume"). In caso contrario la procedura esce per errore.

A parte questa attenzione iniziale, l'utente deve poi solo rispondere alle domande, peraltro estremamente chiare, che gli vengono rivolte dal calcolatore.

Esempio (solite convenzioni)

(Supponiamo che inizialmente il nastro standard di nome TROPPI
contenga 5 files, chiamati nell'ordine: PRIMO.DAT;1,
SECONDO.DAT;4, TERZO.DAT;6, QUARTO.DAT;3, QUINTO.DAT;4)

libera_nastro troppi

(appare sulla console il messaggio :

MONTA IL NASTRO TROPPI PER CANCELLAZIONE DI FILES

la procedura si ferma in attesa che l'operatore fornisca la
risposta)

NOME COMPLETO DEL FILE DA CANCELLARE : terzo.dat;6

FILES DA CANCELLARE :

TERZO.DAT;6

QUARTO.DAT;3

QUINTO.DAT;4

O.K.? : si

(la procedura esegue le ultime operazioni e poi restituisce il
controllo al DCL)

Sul nastro TROPPI restano solo due files : PRIMO.DAT;1 e

SECONDO.DAT;4 .

USO DEL "BATCH" E DELLE CODE DI STAMPA

E' noto che il Vax e' un elaboratore orientato soprattutto per l'esecuzione di lavori interattivi e per il governo di processi in tempo reale. Tuttavia il sistema operativo e' attrezzato anche per l'esecuzione di lavori "batch", la cui organizzazione, come si sa, e' propria piu' di un "mainframe" che di un mini. In questo senso il Vax puo' essere classificato come un "supermini", cioe' a meta' strada fra i mini tradizionali, che eseguono solo lavori interattivi, e i grossi elaboratori, attrezzati con tutte le "facilities" di supporto all'elaborazione "batch".

A chi non e' ancora familiarizzato con questi concetti diciamo che un lavoro "batch" e' eseguito dall'elaboratore in modo asincrono e non-interattivo rispetto alla chiamata dell'utente: questi invia all'elaboratore una procedura, detta "Job", contenente comandi DCL, programmi e dati; il Job viene sistemato in una coda di attesa e viene eseguito quando e' possibile in funzione del numero di Jobs in coda e delle diverse priorit`. Alla fine, il "rapporto" sul Job e gli eventuali risultati mandati in stampa dalla procedura vengono posti in una coda di uscita e, ancora quando e' possibile, stampati. E' chiaro quindi che l'utente non puo' interagire con l'esecuzione del lavoro, dovendosi limitare a "spedire" il Job e a riceverne i risultati. Nel frattempo puo' eseguire altri lavori o puo' lasciare libero il terminale per altri utenti.

Il modo "batch" e' particolarmente indicato per lavori che utilizzano le risorse della macchina piuttosto pesantemente, soprattutto per quello che riguarda il tempo di esecuzione. Da que-

sto punto di vista lavorare in "batch" conviene non solo all'utente, ma anche all'organizzazione generale: se la stessa procedura fosse svolta interattivamente, il terminale resterebbe bloccato nella lunga attesa che il programma completi la sua esecuzione; e, si sa, i terminali sono sempre pochi rispetto alle esigenze degli utenti!

Abbiamo visto che condizione per la possibilita' di fare lavoro "batch" e' l'esistenza di almeno due code, una di input per i jobs in attesa di esecuzione e una di output per le stampe. Le code sono definite e gestite dal responsabile del centro (o dall'operatore) con particolari comandi DCL; questi comandi, eseguibili solo con certi privilegi, svolgono diverse funzioni, quali la creazione di una coda, la sua cancellazione, l'attivazione o l'arresto di una coda, il blocco di un file in una coda, il suo rilascio, il cambiamento di prioritaa' fra i files all'interno di una coda ecc...; il criterio seguito deve essere quello di ottimizzare il funzionamento generale, tenendo conto del carico di utenza, delle risorse del sistema e del numero e del tipo di periferiche disponibili. Il responsabile puo' anche stabilire che siano definite piu' code, sia in input che in output e puo' fissare il numero di jobs in una coda batch che possono essere mandati in esecuzione contemporaneamente; per esempio, ci potrebbero essere due code batch con diverse prioritaa' e diverso numero di jobs ammessi in esecuzione contemporaneamente: in quella a prioritaa' piu' alta andrebbero a finire i jobs veloci e che occupano meno risorse, nell'altra i jobs con caratteristiche contrarie; oppure ci potrebbero essere tante code di stampa quante sono le stampanti e cosi' via. Le due code di "default" sono la SYS\$BATCH per

l'input e la SYS\$PRINT per la stampa.

Associata alle code e' la tecnica dello "spooling", che consiste nell'usare un'unita' periferica veloce (il disco) come supporto intermedio nelle comunicazioni fra unita' di input-output lente e l'elaboratore. Se un'unita' lenta e' definita "spooled", i dati sono trasmessi da (o verso) quell'unita' non durante l'esecuzione del programma (la cui velocita' sarebbe cosi' limitata dall'unita' stessa), ma prima (o dopo), mentre l'elaboratore e' impegnato in altri lavori; il programma, quando e' pronto, trova sia' (o mette) i dati su disco, con notevole vantaggio per il tempo di esecuzione. Cosi' la trasmissione di dati fra le unita' "spooled" (come per esempio il "card-reader" e le stampanti) e il disco avviene in modo completamente asincrono rispetto all'esecuzione dei programmi.

Se il centro dispone di un "card-reader" (non e' il caso nostro, per il momento) e se questo, come avviene sempre, e' definito "spooled", un modo possibile per inviare jobs nelle code batch e' quello di usare le schede perforate, premettendo al pacco che contiene la procedura le due schede \$JOB e \$PASSWORD.

Un altro modo per inviare un job nelle code batch e' quello di usare il comando SUBMIT seguito dal nome del file che contiene il job stesso. Questo comando e' particolarmente utile in quanto puo' essere inviato da terminale interattivo. L'utente e' vivamente presato di studiare bene tutte le possibilita' del comando SUBMIT, la cui conoscenza e' condizione necessaria per poter lavorare proficuamente in modo "batch". Per una spiegazione in dettaglio dell'uso dei comandi \$JOB, \$PASSWORD e SUBMIT, vedere il manuale della Digital: VAX/VMS COMMAND LANGUAGE USER'S GUIDE.

Eseguito il Job, il sistema crea il file dei risultati a cui da', come nome, lo stesso della procedura e, come tipo, il simbolo LOG. Per esempio, se la procedura si chiama MYJOB.COM e l'utente invia il comando SUBMIT MYJOB, il file dei risultati si chiamera' MYJOB.LOG (nel directory di "default" dell'utente). Lo stesso file sara' disposto per la stampa nella coda SYS\$PRINT. Se a sua volta la coda SYS\$PRINT e' associata a un'unita' definita "spooled", la stampa verra' effettivamente eseguita appena possibile; altrimenti il file rimane in "hold" finche' l'operatore non assegnera' il simbolo SYS\$PRINT a un'unita' fisica. Terminata la stampa, il file dei risultati e' automaticamente cancellato.

SITUAZIONE ATTUALE

In attesa della ormai prossima installazione di un lettore di schede e di 2 stampanti, la presente configurazione e' ancora tale da sacrificare notevolmente l'uso del "batch": 512K di memoria sono troppo pochi per consentire piu' di una coda di input, mentre una sola telescrivente (la LA120 collegata all'uscita TTA0:), forzata a lavorare come stampante, e' troppo debole per riuscire a smaltire una grossa coda di stampa. Per cui la linea adottata finora e' stata quella di mantenere la configurazione "batch" a un livello minimale, e cioe' costituita da:

- 1) Una sola coda di input, di nome SYS\$BATCH, con un solo Job per volta in esecuzione. Questa e' la stessa coda di "default" a cui indirizza i Jobs il comando SUBMIT, il quale resta, fra l'altro, l'unico mezzo attualmente disponibile per l'esecuzione

ne di lavori "batch".

2) Due code di stampa e precisamente:

a) la coda TTA0, associata all'omonima unita' fisica: i files indirizzati a questa coda sono posti in attesa e poi stampati sulla telescrivente. Questa coda e' usata soprattutto per produrre stampe provenienti da lavori interattivi (vedi piu' avanti il comando PRINT)

b) la coda SYS\$PRINT, attualmente non assegnata a unita' fisiche. In questa coda vengono inviati per "default" tutti i "files.LOG", cioe' le stampe prodotte da lavori batch. La stampa effettiva, pero', non avviene, se non quando l'operatore, dietro richiesta dell'utente interessato, scarica il contenuto della coda su TTA0.

La linea di non automatizzare le stampe prodotte da lavori batch (cioe' in modo non direttamente controllabile dall'utente) e' stata scelta per non appesantire inutilmente il lavoro dell'unica stampante (telescrivente) disponibile: l'utente puo' controllare il suo "file.LOG" e decidere a posteriori se vale la pena o no di stampare i risultati. Nel caso affermativo chiede all'operatore di scaricare la coda logica SYS\$PRINT (o solo il suo file) sulla coda fisica TTA0. Se invece decide di cancellare il file, deve attenersi a questa sequenza di operazioni:

- 1) Cancellazione del "file.LOG", tramite comando DELETE.
- 2) Lista, con il comando SHOW QUEUE/FULL SYS\$PRINT, del contenuto della coda; da tale lista l'utente ricava il "job-number" n dell' "entry" che corrisponde al suo file.
- 3) Cancellazione dell' "entry" n con il comando :
DELETE/ENTRY=n SYS\$PRINT.

COMANDO PRINT

Il comando PRINT (forme abbreviate: PRIN,PRI,PR) e' stato ridefinito: nel suo significato originario questo comando inviava un file in stampa nella coda SYS\$PRINT. L'attuale comando corrisponde invece a : PRINT/QUEUE=TTA0 , cioe' accoda il file in TTA0.

In questo modo si e' voluto separare l'interattivo e il batch su due strade diverse: le stampe prodotte in interattivo, e per questo piu' facilmente controllabili, vanno direttamente sulla coda fisica, mentre le altre restano "congelate" nella coda SYS\$PRINT, permettendo cosi' che un esame del loro contenuto decida se farle proseguire o cancellarle.

In ogni caso, all'utente "fiducioso in se' stesso" resta comunque la possibilita' di stabilire a priori che la stampa del suo Job sia eseguita direttamente: basta che assegni l'unita' fisica TTA0 al nome logico SYS\$PRINT, cioe' inserisca in testa alla procedura il comando: ASSIGN TTA0: SYS\$PRINT.

Come conseguenza di questo comando il "file.LOG" prodotto dal Job sara' inviato in stampa su TTA0.

NUOVI COMANDI DI VARIA APPLICAZIONE

Sono stati definiti alcuni simboli globali che, usati come comandi, svolgono la stessa funzione di comandi DCL connessi a particolari qualificatori, oppure mandano in esecuzione procedure o programmi. A questa categoria appartengono i comandi : REWIND, SKIP, DA_NASTRO, SU_NASTRO, LIBERA_NASTRO e PRINT, sia' descritti in dettaglio nei capitoli precedenti.

Restano da esaminare i seguenti comandi:

CODICE

Il comando CODICE serve solo ai programmatori che sanno usare in modo sofisticato le routines di sistema, chiamate 'System Services'. Il comando manda in esecuzione una procedura che stampa due liste, una in ordine alfabetico e l'altra in ordine numerico, in cui sono elencati tutti i codici di sistema di un certo tipo, insieme ai loro corrispondenti valori numerici esadecimali. Conoscere questa corrispondenza e' utile perche', mentre sui manuali i codici sono sempre riferiti tramite il loro nome simbolico, il programmatore puo' aver bisogno qualche volta di usare i valori numerici, per esempio per preparare certe maschere.

Per ottenere questo, l'utente deve inviare il comando CODICE insieme al nome del tipo dei codici che vuole listare; tale nome deve essere scritto nella stessa forma in cui e' chiamata la 'macro' corrispondente, a parte la desinenza DEF che e' usuale tutti.

Per esempio, se si vuole conoscere il codice numerico dei messaggi di errore di sistema (e viceversa), la macro corrispondente e' SSDEF; per cui bisogna inviare il comando: CODICE SS

Per i codici di input-output (macro IODEF), bisogna battere: CODICE IO ecc...

CONVERTI

Il comando CONVERTI manda in esecuzione un programma che converte numeri dalla forma esadecimale a quella decimale e viceversa. I numeri sono letti, convertiti e scritti uno alla volta, finche' l'utente non batte un CTRL Z (end-of-data) provocando cosi' la fine del programma.

Istruzioni per l'uso piu' dettagliate sono contenute in un breve testo che appare sul terminale non appena si invia il comando e prima della lettura del primo dato da convertire.

EDT

Il comando EDT sta per EDIT/EDT e fa si' che il controllo passi a questo nuovo tipo di Editor, che si presenta molto interessante, soprattutto per la possibilita', assai comoda, di "editare" correndo direttamente il testo sull'intero schermo.

L'unica difficolta' sta nel fatto che l'EDT, non essendo una immagine installabile, assorbe notevoli risorse al sistema; per cui si sconsiglia di usare tale comando quando il sistema risulta troppo carico.

EXTLNK

Il comando EXTLNK chiama una procedura che esegue il comando LINK, associando al programma dell'utente la libreria: SYS\$LIBRARY:STARLET, membro LIB\$ESTEMU.

Il comando va usato quando il programma da far leggere al "linkase-editor" e' stato compilato con l'opzione G_FLOAT (dati REAL*8 con il "range" tipo CDC) oppure contiene dati REAL*16 (doppia precisione tipo CDC).

PRECESSIONI ASTRONOMICHE

(il presente lavoro e' stato eseguito con la collaborazione
di L. Gregorini)

Sono state scritte una procedura e alcune subroutines che eseguono il calcolo di posizioni astronomiche ad una certa data, corrette per la precessione, partendo da posizioni note al 1950.0 (o viceversa). Procedura e subroutines risiedono nel directory [ASTRO].

PRECIN

La procedura PRECIN esegue il calcolo di una o piu' posizioni astronomiche corrette per la precessione.

Prima di mandarla in esecuzione e' conveniente definire il file di stampa, tramite il comando: ASSIGN file FOR006 , dove "file" e' il nome di un file dell'utente oppure il nome della coda di stampa "TTA0:". Nel caso non si batta questo comando, la stampa dei risultati appare sullo schermo del terminale e non puo' essere conservata.

Ordine di chiamata :

@[ASTRO]PRECIN (senza parametri)

La procedura chiede:

- 1) il nome del file di input (posizioni da precessare)
- 2) il nome del file di output (posizioni precesstate)
- 3) l'anno (ultime 2 cifre) dei "Besselian Day Numbers" da usare
(per ora sono disponibili solamente quelli del 1981)

infine appare la scritta :

PROGRAMMA PRECIN IN ESECUZIONE

Il programma, mandato in esecuzione dalla procedura omonima, legge da terminale, nell'ordine, i seguenti dati:

- 1) Una stringa alfanumerica, racchiusa fra parentesi tonde, che descrive il formato con cui sono scritti i dati nel file di input e con cui verranno scritti i risultati nel file di output.
- 2) Una stringa alfanumerica, racchiusa fra parentesi tonde, che descrive il formato con cui sarà scritto il file di stampa.
- 3) Quattro numeri interi [FORMAT (4I3)] che rappresentano giorno, mese, anno (2 cifre) della data a cui si vuol riferire la precessione e un "flag" che deve essere +1 se si vuole precessare dal 1950.0 in avanti, e -1 se si vuole precessare in direzione contraria.

Per quanto riguarda i formati di lettura e scrittura, il programma presuppone che sia i dati di input che quelli di output e di stampa siano costituiti da variabili disposte in questa sequenza:

C1, IO, MI, ST, C2, IG, IP, SS, C3 dove:

C1, C2 e C3 sono campi alfanumerici di almeno 1 e di al massimo 80 caratteri ciascuno (formato A); questi campi corrispondono a tre stringhe di dati che vengono trasferite senza modifiche nei files di output e di stampa.

Le altre variabili sono intere (formato I), salvo ST e SS che devono essere in formato F. Il loro significato è il seguente:

IO, MI, ST sono: ore, minuti e secondi dell'ascensione retta

IG, IP, SS sono: gradi, primi e secondi d'arco della declinazione

(se la declinazione è negativa, allora anche IP e SS devono essere forniti col segno -)

Terminata l'esecuzione del programma, la procedura torna a chiedere il nome di un nuovo file di input, e così via, finché l'utente non interrompe il "loop", battendo una risposta nulla (carriage-return).

SUBROUTINES PER LA PRECESSIONE

Il programma PRECIN chiama le subroutines PRINIZ, PRPOS, PRDAY e PREXEC, che costituiscono 4 "entries" diversi dello stesso modulo, di nome PRECES .

Le subroutines sono utilizzabili da un qualunque programma che esegua calcoli di precessioni. Se il programma si chiama, per esempio, MAIN, il modulo PRECES deve essere inserito nel comando LINK con la seguente chiamata:

```
LINK MAIN,[ASTRO]PRECES
```

PRINIZ

La subroutine PRINIZ inizializza l'operazione di calcolo delle precessioni, leggendo i "Besselian Day Numbers" dall'unità logica 9. Quindi, a meno che non si usi la procedura PRECIN (nel qual caso l'assegnazione è automatica) l'utente deve assegnare il file che contiene i BDN ([ASTRO]BES1981.DAT) al nome logico FOR009.

Chiamata:

```
CALL PRINIZ (NAN)         dove:
```

NAN, restituito dalla subroutine al programma chiamante, contiene le ultime due cifre dell'anno a cui si riferiscono i BDN.

PRPOS

La subroutine PRPOS riceve dal programma chiamante la posizione su cui calcolare la precessione.

Chiamata:

CALL PRPOS(IO,MI,ST,IG,IP,SS) dove:

IO,MI,ST e' l'ascensione retta in ore, minuti e secondi

IG,IP,SS e' la declinazione in gradi, primi e secondi d'arco.

PRDAY

La subroutine PRDAY riceve dal programma chiamante la data a cui riferire la precessione.

Chiamata:

CALL PRDAY(J,M) dove:

J e' il giorno

M e' il numero del mese.

PREXEC

La subroutine PREXEC esegue il calcolo della precessione restituendo la posizione corretta.

Chiamata:

CALL PREXEC(IND,IOP,MIP,STP,IGP,IPP,SSP) dove:

IND e' un numero fornito alla subroutine (+1 o -1), che indica il verso della precessione

Le altre variabili sono ascensione retta e declinazione precessate, di ritorno al programma chiamante.

La subroutine PRINIZ deve essere chiamata per prima e la PREXEC deve essere chiamata per ultima.

PRPOS e PRDAY possono essere chiamate in qualunque ordine a seconda della necessita' del programma. Per esempio, se serve precessare un intero catalogo alla stessa data, conviene chiamare la PRDAY una sola volta e poi la PRPOS e la PREXEC tante volte quante sono le sorgenti. Se invece si vuole conoscere le diverse posizioni nell'arco dell'anno di una stessa sorgente, conviene chiamare prima la PRPOS con le coordinate della sorgente e poi la PRDAY e la PREXEC tante volte quante sono le date a cui si vuole precessare.

LIBRERIA DI PROGRAMMI PER ESPERIMENTI VLBI

Esiste un primo nucleo di libreria per esperimenti VLBI (Very Long Baseline Interferometers). Attualmente i programmi disponibili sono soltanto i seguenti: VLBFIT, UVFLOT, CORTEL, INVERT e CLEAN, ma la procedura per il trattamento VLBI e' in via di sviluppo e la libreria e' in continuo ampliamento.

VLBFIT

Il programma VLBFIT, partendo da un dato modello iniziale e considerando solo le ampiezze di visibilita', seleziona il modello di radiosorvente che meglio si adatta ai dati sperimentali.

Il programma originale e' stato scritto da R.T. Schilizzi e G. Purcell; ad esso sono stati apportati alcuni miglioramenti, quali il calcolo della cosiddetta "closure-phase" per il modello e le osservazioni, che permette una migliore verifica dell'accordo fra modello e dati sperimentali.

Funziona in modo completamente interattivo e produce anche grafici di confronto fra dati e modello.

La versione attuale e' stata migliorata e adattata da C. Fanti.

UVPLOT

Il programma UVPLOT produce il grafico della distribuzione dei punti osservati sul piano UV. Può essere di valido aiuto nella prima fase di analisi dei dati e nella formulazione di un approssimato modello iniziale.

Il programma è stato fatto da M. Massi e riadattato per il Vax da L. Padrielli.

CORTEL + INVERT + CLEAN

Questi programmi operano insieme iterativamente e permettono di produrre mappe delle sorgenti, usando entrambe le informazioni sull'ampiezza di visibilità e sulla fase, senza passare attraverso il metodo della "closure-phase".

Il metodo è stato sviluppato da Cornell e Wilkinson e riadattato per il Vax da C. Fanti.

La procedura usata è ancora in via di messa a punto e non è da considerarsi del tutto operativa.

Le istruzioni (in inglese) per l'uso di questi programmi sono in un file, di nome: [CARLA]MANUAL.VLB .

STARLINK : UN ESEMPIO DELL'USO DI ELABORATORI VAX

IN APPLICAZIONI ASTRONOMICHE

L'astronomia inglese si e' recentemente attrezzata con un enorme sistema di elaborazione dati, costituito da 6 elaboratori Vax, situati nei principali centri di ricerca e collegati fra loro in rete.

Il progetto che ha portato alla realizzazione di questo sistema, di nome STARLINK, e' stato finanziato e gestito dallo "Science Research Council" (qualcosa di simile al nostro CNR), allo scopo di fornire alla comunita' astronomica i mezzi per risolvere i nuovi problemi di analisi dei dati e di elaborazione delle immagini che derivano dalle aumentate capacita' e potenzialita' degli strumenti.

Fino a non molto tempo fa, in Inghilterra, come in ogni altra parte del mondo, le attrezzature a disposizione degli astronomi per l'analisi delle loro osservazioni erano molto limitate. I ricercatori dovevano o ridurre i dati manualmente o costruirsi le attrezzature di hardware e software "ad hoc" per il loro esperimento, con conseguente polverizzazione e spreco di energia, a scapito dell'efficienza e dell'interesse generale. Nel contempo si faceva sempre piu' forte l'esigenza di uniformare tutto questo lavoro disperso e di adeguare i mezzi di analisi alla crescita vertiginosa delle potenzialita' di acquisizione degli strumenti.

Infatti in questi ultimi anni si sono sempre piu' raffinate le tecniche di acquisizione e di conversione di dati sotto forma digitale, operate, ad alto potere risolutivo, da macchine automatiche sempre piu' veloci e precise, che, o lavorano in linea con i telescopi, o analizzano le lastre fotografiche, producendo, in un caso o nell'altro, un'enorme quantita' di dati numerici.

Nell'immediato futuro, il lancio di satelliti attrezzati per l'osservazione astronomica (basti fra tutti ricordare l'ormai prossimo Space Telescope), aumentera' in maniera drammatica questa produzione di dati, il cui contenuto scientifico potra' essere adeguatamente analizzato solo con potenti e sofisticati mezzi di calcolo.

Inoltre, sempre a causa dell'enorme numero di dati da elaborare insieme, si e' sentita molto forte l'esigenza di disporre di strumenti atti all'analisi interattiva dell'immagine, meglio se con l'uso del colore che permette agevolmente di distinguere zone di diversa densita'. Infatti l'esperienza ha gia' dimostrato che un'opportuna combinazione di potenza di calcolo e interazione visiva con i dati riconvertiti in immagini, accresce enormemente l'efficienza e la velocita' con cui importanti risultati scientifici possono essere ottenuti.

Era quindi necessario disporre di elaboratori che fossero non solo potenti, ma anche e soprattutto orientati per l'elaborazione interattiva.

Sensibilizzato a questi problemi, lo Science Research Council (e qui le analogie con il nostro CNR finiscono) ha promosso e velocemente realizzato il progetto STARLINK, orientando sul Vax la scelta dell'elaboratore che meglio soddisfa alle esigenze suddette.

Ciascuno dei 6 centri, collegati fra loro attraverso una rete telefonica a 9600 baud, dispone di: 2 Mbyte di memoria fisica, 4 unita' disco da 176 Mbyte, 2 unita' nastro, 2 unita' video per il processo delle immagini e un numero imprecisato di plotters, stampanti e terminali interattivi.

Il software, specializzato per problemi di elaborazione astronomica, e' prodotto dai ricercatori stessi, ma viene coordinato, reso omogeneo e ridistribuito in un unico centro. Una particolare attenzione e' stata posta nella realizzazione di un linguaggio di comandi generalizzato e di facile uso, mediante il quale l'utente, dovunque si trovi, puo' agevolmente invocare l'esecuzione di un programma.

Davanti a un simile colosso, il nostro centro non si puo' che definire "microscopico". Tuttavia non e' escluso che in futuro si possa trarre buon frutto dalla loro esperienza e che un po' della loro metodologia (e dei loro programmi) possa essere applicata anche al nostro Vax. In questo senso gia' ci sono stati dei contatti informali fra alcuni ricercatori dell'Istituto di Radioastronomia e i colleghi inglesi; si spera che in futuro tali contatti possano essere sempre piu' frequenti, nel comune interesse della ricerca in astronomia.

PROSSIMAMENTE

- 1) E' urgente risolvere il problema della quota di spazio-disco assegnata a ogni utente. Infatti il numero di files degli utenti cresce continuamente, mentre l'unita' disco resta unica, e lo restera' ancora per molti mesi. Il problema e' stato risolto finora assegnando una quota molto limitata ad ogni utente e consigliando di salvare tutto il possibile su nastro, soprattutto per quello che riguarda i files che non sono usati quotidianamente. Nonostante questo accorsimento, il problema si pone ancora, in special modo quando l'utente crea, all'interno del proprio Job, dei files temporanei o delle stampe particolarmente lunghe. Infatti anche i files temporanei e i 'files.LOG', nonostante la loro provvisorieta', sono contati nella quota, superata la quale il Job abortisce senza curarsi del fatto che alla fine tali files saranno cancellati. La linea che si seguira' sara' probabilmente quella di scrivere un programma che, utilizzando certe routines di sistema, sara' in grado di assirare il controllo della quota nel caso di files definiti temporanei. Ovviamente tali files saranno cancellati automaticamente una volta trascorso un certo limite di tempo.

- 2) I programmi che applicano la tecnica di 'CLEAN AND RESTORE' alle registrazioni del radiotelescopio di Medicina, funzionanti gia' da alcuni anni sull'IBM del CNUCE, sono stati riadattati per l'uso sul Vax. In aggiunta sono state fatte alcune procedure di comandi DCL che permettono all'utente di costruire interattivamente i pacchetti di dati input per questi pro-

grammi, i quali invece, data la loro complessità, funzionano in modo "batch". Il prossimo numero del rapporto conterra le dettagliate istruzioni per l'uso dei programmi e delle procedure.

- 3) Sono in corso di sistemazione e di riadattamento per il Vax i programmi di elaborazione dei dati raccolti con il radiotelescopio olandese di Westerbork. Questi programmi furono scritti originariamente per elaboratori IBM. Della loro traduzione si stanno occupando L. Feretti e L. Gregorini.
- 4) I ricercatori del "Max Planck Institut Fur Radioastronomie" di Bonn hanno messo gentilmente a nostra disposizione un grosso "package" di programmi di elaborazione dati radioastronomici, scritto per un elaboratore CDC. Il lavoro di riadattamento al Vax è piuttosto arduo, date le fondamentali differenze di struttura fra le parole CDC e Vax. Se ne sta attualmente occupando P. Tomasi.
- 5) Si stanno mantenendo stretti contatti di collaborazione con l' INFN di Bologna che entro brevissimo tempo installerà un Vax presso la sede del CNAF. In particolare, alcune parti della libreria del CERN di Ginevra, riadattate per il Vax, sono state caricate sul nostro elaboratore, grazie anche all'aiuto del Prof. D. Bollini dell'Università di Bologna. Siamo in attesa di procurarci la documentazione necessaria per l'uso di tale libreria.

DOCUMENTAZIONE

Il centro dispone della serie completa di manuali Digital sul VAX/VMS Versione 2. I manuali sono a disposizione degli utenti per consultazione in loco. Chi desidera ottenere dei manuali propri deve acquistarli direttamente dalla ditta, indicando nella lettera di richiesta il titolo e la sigla. L'indirizzo e':
DIGITAL - CENTRO ADDESTRAMENTO - VIA F. TESTI 10 -
20092 CINISELLO BALSAMO (MILANO)

I manuali la cui lettura e' fortemente raccomandata sono i seguenti:

a) per gli utenti che non sanno nulla del Vax:

1) PRIMER

b) per quelli un po' piu' esperti:

- 1) COMMAND LANGUAGE USER'S GUIDE
- 2) GUIDE TO USING COMMAND PROCEDURES
- 3) TEXT EDITING REFERENCE MANUAL
- 4) EDT EDITOR REFERENCE MANUAL

c) per i programmatori fortran:

- 1) FORTRAN IV-PLUS LANGUAGE REFERENCE MANUAL
- 2) FORTRAN IV-PLUS USER'S GUIDE

DOCUMENTAZIONE IN LINEA

Esiste la possibilita' di ottenere la lista su terminale o la stampa di due particolari files:

- 1) Il file [RAPPORTO]INDICE.NAT contiene gli indici dei rapporti nell'ordine in cui questi sono stati scritti.
- 2) Il file [RAPPORTO]INDICE.ALF contiene l'elenco, in ordine alfabetico, dei titoli degli argomenti trattati. Accanto a ogni titolo compaiono tre numeri: il primo indica la pagina, il secondo il numero del rapporto e il terzo l'anno. Esempio:

PRECESSIONI ASTRONOMICHE...../48/1/81

L'argomento PRECESSIONI ASTRONOMICHE e' trattato a Pag.48 del rapporto N.1 del 1981.