



Consiglio Nazionale delle Ricerche

## **Un Sistema di Backup in Ambiente AFS**

**Giancarlo Bartoli**

**Rapporto CNUCE-B4-1998-024**

***CNUCE***

**Pisa**



# **Un Sistema di Backup in Ambiente AFS**

**Giancarlo Bartoli**

**CNUCE – Istituto del CNR**  
Via S. Maria, 36 – 56100 Pisa

e-mail:  
[giancarlo.bartoli@cnuce.cnr.it](mailto:giancarlo.bartoli@cnuce.cnr.it)

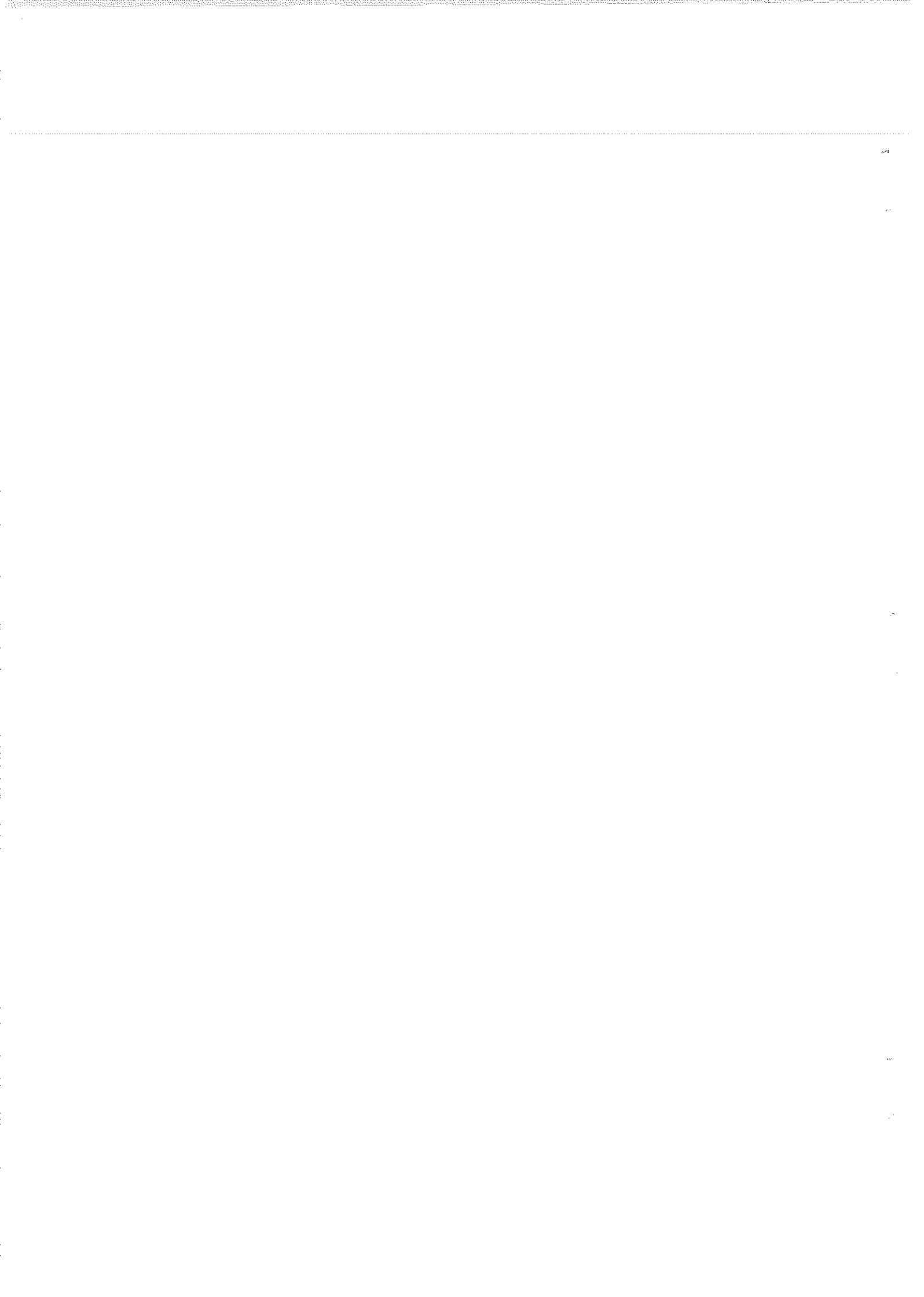


# Indice

|     |   |    |
|-----|---|----|
| 1   | Introduzione .....  | 1  |
| 2   | Introduzione al Sistema di Backup .....                         | 2  |
| 2.1 | Volume Sets e Volume Entries .....                              | 2  |
| 2.2 | Dump e Dump set .....   | 2  |
| 2.3 | La gerarchia e i livelli del dump .....                         | 3  |
| 2.4 | I nomi dei dump e dei nastri .....                              | 3  |
| 2.5 | Data di scadenza .....  | 4  |
| 2.6 | Recupero dei dati salvati .....                                 | 4  |
| 2.7 | Tape Labels, Dump Labels e EOF Markers .....                    | 4  |
| 2.8 | Macchina Tape Coordinator .....                                 | 5  |
| 2.9 | Il Database del Backup .....                                    | 5  |
| 3   | Configurazione del Sistema di Backup .....                      | 7  |
| 3.1 | Determinazione della capacità del nastro e del EOF Marker ..... | 7  |
| 3.2 | Configurazione della macchina Tape Coordinator .....            | 8  |
| 3.3 | Aggiunta o Rimozione di una macchina Tape Coordinator .....     | 9  |
| 3.4 | Definizione dei Volume Sets e Volume Entries .....              | 10 |
| 3.5 | Creazione della gerarchia del dump .....                        | 13 |
| 3.6 | Definizione della data di scadenza per i Dump Levels .....      | 13 |
| 3.7 | Inizializzazione di un nastro .....                             | 14 |
| 4   | L'interfaccia del Sistema di Backup .....                       | 17 |
| 4.1 | La finestra del Tape Coordinator .....                          | 19 |
| 5   | Visualizzazione del Backup .....                                | 21 |



|     |   |    |
|-----|---|----|
| 6   | Salvataggio dei dati  | 26 |
| 6.1 | Il processo di Backup   | 26 |
| 6.2 | Uso dei nastri con l'operazione di Backup                             | 27 |
| 6.3 | Mettere multipli dump in un dump set                                  | 28 |
| 7   | Procedure di automatizzazione del Sistema di Backup                   | 30 |
| 7.1 | Disabilitare la richiesta per l'inserimento del nastro                | 30 |
| 7.2 | Schedulazione automatica dei Dumps                                    | 30 |
| 8   | Recupero dei dati   | 33 |
| 8.1 | Specifiche del tipo e della destinazione di una operazione di restore | 33 |
| 9   | Procedura di Backup nel RCSA  | 36 |
| 10  | Appendice A   | 39 |
| 11  | Bibliografia  | 45 |



## **Prefazione**

Nel panorama scientifico e tecnologico italiano il CNUCE rappresenta un punto di riferimento per quanto concerne la ricerca informatica, l'organizzazione e l'erogazione di servizi informatici e telematici ad alto valore aggiunto. Nel tempo l'Istituto ha maturato importanti competenze sull'utilizzo dell'Informatica in settori disciplinari ed esistono, pertanto, sempre più forti opportunità di mettere a disposizione del mondo scientifico, industriale e della Pubblica Amministrazione le competenze sviluppate, svolgendo così anche un importante ruolo sociale. Queste opportunità, insieme a quanto appare nel panorama mondiale, hanno reso inevitabile un ripensamento sulla missione e sulla evoluzione del vecchio Servizio di Calcolo del CNUCE, verso una nuova struttura intesa come organizzazione di competenze e strumentazione per erogare servizi specialistici integrati con le attività scientifiche dell'Istituto o strettamente correlati con esse, in settori che ricadono prioritariamente sotto il dominio del Comitato per le Scienze di Ingegneria ed Architettura (07). Per questo è stato creato il Reparto Calcolo Scientifico Avanzato (RCSA) che è quindi inteso come organizzazione di competenze e strumentazione HW/SW per erogare verso l'esterno dell'Istituto servizi specialistici integrati con le attività scientifiche dell'Istituto o correlati con esse.

La tipologia dei servizi offerti da RCSA è la seguente:

- servizi offerti ad uno o più Reparti per il supporto ad attività progettuali, consistenti nell'attivazione di sistemi informatici e nella loro gestione per le attività proprie del progetto;
- offerta di software specialistico nei settori di competenza dei Reparti;
- offerta di software specialistico nell'ambito di collaborazioni, dove uno o più Reparti possano offrire un contributo scientifico tecnologico rilevante;
- consulenza specialistica nei settori di competenza dei Reparti;
- attività di formazione nei settori di competenza dei Reparti;
- gestione, per conto dell'Ente, di convenzioni da attuarsi tra la Comunità scientifica nazionale e fornitori di software specialistici, per la promozione di competenze e dei relativi strumenti.

---

## 1. Introduzione

Il problema della protezione dei dati è stato per molto tempo sottovalutato: ci si accorge della loro vulnerabilità solo quando si manifesta un problema e l'integrità dei dati è ormai compromessa. A tale riguardo, il backup è diventato un fattore essenziale per lo sviluppo informatico sia in ambito pubblico che privato e quindi proteggere i dati importanti diventa una funzione vitale per qualunque attività. La perdita dei dati porta infatti a conseguenze disastrose sia di carattere economico che tecnico. Una caratteristica fondamentale del backup è dunque la garanzia dell'affidabilità: un requisito base da soddisfare prima ancora della potenza di calcolo o di altri parametri del sistema poiché, in questo modo si ha il vantaggio di avere un minor tempo d'inattività a seguito di un problema.

La spesa decisamente modesta e l'affidabilità che caratterizzano i sistemi di backup sono quindi degli elementi che dovrebbero convincere ogni azienda od ente a dotarsi di dispositivi di questo tipo, che risultano essere sempre più delle componenti fondamentali per la qualità e l'affidabilità di una qualsiasi struttura.

Alla luce dell'importanza delle funzioni offerte dal Reparto Calcolo Scientifico Avanzato, un primo principale compito del Reparto è stato quello di assicurare l'integrità fisica dei dati degli utenti mediante il salvataggio ed un eventuale ripristino dopo perdite accidentali.

La finalità del lavoro presentato in questo rapporto è quella di descrivere in dettaglio un Sistema di Backup che è stato sviluppato per l'ambiente AFS al fine di garantire agli utenti l'integrità delle informazioni in caso dei possibili ed eventuali guasti hardware e software. Sono inoltre riportate le modalità di attivazione di tale sistema per il salvataggio dei dati degli utenti utilizzatori delle risorse del Reparto Scientifico Avanzato (RCSA).

---

## 2. Introduzione al Sistema di Backup

Il Sistema di Backup AFS permette di ottenere due principali funzioni:

- Fare il salvataggio dei dati AFS in un nastro.
- Recuperare nel filesystem AFS i dati salvati sul nastro.

Il sistema di Backup di AFS permette di controllare molti aspetti del processo di salvataggio, quali il numero dei salvataggi da eseguire, quali volumi devono essere salvati e quanto spesso le versioni parziali o totali devono essere eseguite.

### 2.1 Volume Sets e Volume Entries

Il Backup dei dati del filesystem AFS prevede di definire un volume base e, prima che il suo contenuto sia salvato, il volume deve esser associato ad un *volume set*. Un *volume set* è un gruppo di uno o più volumi che sono salvati su un singola unità nastro. Se un volume non è associato con la definizione di *volume set*, i dati che esso contiene non possono essere salvati con il Sistema di Backup di AFS.

L'associazione dei volumi ad un *volume set* avviene definendo uno o più *volume entry* per quel volume. Un *volume entry* identifica il nome e la locazione di uno o più volumi usando tre campi: il nome della macchina *file-server*, il nome della partizione ed il nome del volume. Tutti i campi supportano i metacaratteri così che un volume può essere usato per identificare un singolo volume o un gruppo di volumi con nomi simili.

Ai fini del backup, i volumi di un *volume set* sono trattati come una singola unità. Essi sono salvati insieme e memorizzati sullo stesso nastro o set di nastri.

### 2.2 Dump e Dump set

Un *dump* è un singolo salvataggio di un singolo *volume set*. Ci possono essere i seguenti tipi di dump:

- Un *full dump* è una copia completa di tutti files nello specificato *volume set*.
- Un *incremental dump* è una copia di quei files nello specificato *volume set* che hanno subito una modifica da quando è stato eseguito un precedente *parent dump*.
- Un *parent dump* è un punto di riferimento per i dump incrementali ed è usato per determinare quali files in un *volume set* devono essere salvati e quali no.

Un *dump set* è un gruppo di uno o più dump posti in uno o più nastri; i nastri in un *dump set* sono identificati usando l'informazione contenuta nel primo dump del *dump set*. Un *dump set* è creato quando un dump è salvato in un nuovo nastro o un nastro contiene un altro *dump set* che è sovrascritto durante il processo di dump.

Una volta che un *dump set* è creato, altri dump possono essere aggiunti. Il primo dump in un *dump set* è referenziato come il *dump iniziale*. Ogni susseguente dump aggiunto nel *dump set* è chiamato *appended dump*. Il record nel *Database di Backup* che identifica il dump aggiunto non può essere cancellato direttamente, per rendere possibile questa operazione, è necessario cancellare l'intero dump.

### 2.3 La gerarchia e i livelli del dump

La *gerarchia del dump* è definita dai livelli che lo compongono; un *dump level* è specificato dal suo *pathname*. In generale, la gerarchia del dump è una schedulazione che indica la frequenza ed il tempo in cui i dump devono essere eseguiti. La gerarchia del dump definisce la relazione che esiste tra il *full*, *incremental* e *parent dump level*. Il *top level* nella gerarchia del dump, rappresenta sempre un *full dump*, ogni livello più basso è un *dump incremental* che fa riferimento al livello più alto. Il seguente esempio mostra un tipo di gerarchia del dump:

```
/sunday1
    /monday
    /tuesday
    /wednesday
    /thursday
    /friday
/sunday2
    /monday
    /tuesday
    /wednesday
    /thursday
    /friday
```

In questo caso, */sunday1* e *sunday2* rappresentano un *full dump*; */sunday1/monday*, */sunday1/tuesday*, */sunday1/wednesday*, */sunday1/thursday*, */sunday1/friday* sono *dump incremental* che dipendono da *sunday1*. Ogni *dump level* deve essere univoco.

### 2.4 I nomi dei dump e dei nastri

Un dump viene identificato sia con il nome del *volume set* salvato che con il nome dell' elemento più basso del *dump level*.

<*volume set name*>. <*dump level name*>

Per esempio, un dump del *volume set user* con il *dump level /sunday/friday* viene identificato come *user.friday*.

Poichè i nomi dei dump non sono univoci, ad ogni dump viene associato un univoco ID usando il numero di secondi dal 12:00 a.m. del 1 Gennaio 1970 (tempo UNIX equivalente a 0).

Il nome dei nastri che contengono un *dump set* è formato dal nome del dump iniziale, dal dump ID e dalla posizione numerica del nastro nel *dump set*, come segue:

<*volumes set name*>. <*dump level name*>.<*tape index*> (*Dump ID*)

Per esempio, se il *volume set uservol* è salvato al *dump level /sunday/monday* e il risultante ID è 123455555, il primo nastro nel dump viene referenziato come: *uservol.monday.1 (123455555)*

## 2.5 Data di scadenza

Quando si crea un *dump level* viene assegnata una data di scadenza; la data di scadenza viene assegnata a tutti dump che sono salvati con lo stesso *dump level*.

La data di scadenza previene le sovrascritture accidentali e può essere specificata usando una data assoluta (June 5, 1997 at 23:59:59) o relativa come per esempio: tre settimane dalla data del salvataggio. La data di scadenza associata con un *dump level* può essere cambiata; ma una volta che è assegnata ad un dump essa non può essere più cambiata per quel dump.

Il solo modo per sovrascrivere un nastro associato con un *dump set* con data non scaduta, è quello di inizializzare di nuovo la label magnetica del nastro e in questo modo saranno rimossi dal *Database di Backup* tutti i record dell'intero *dump set* compresi i dump aggiunti. La data di scadenza di un iniziale dump in un *dump set* è memorizzata nella label del nastro, mentre la data di scadenza dei dump aggiuntivi è memorizzata nel *Database di Backup*.

## 2.6 Recupero dei dati salvati

Una volta che i dati sono salvati su un nastro usando il Sistema di Backup di AFS, questi possono essere recuperati selezionando :

- Uno specifico volume o più volumi.
- Una intera partizione di disco.

Ci sono due modi per recuperare i volumi specificati:

- Un **full restore**                      ricrea il volume come esso era prima del suo salvataggio.
- Un **date-specific restore**        ricrea il volume come esso era alla data specificata.

Una partizione disco può essere recuperata solo se il *backup* dei volumi risiedono nello stesso nastro. Quando si recupera una partizione, il Sistema di Backup consulta il VLDB per determinare quali volumi sono coinvolti.

## 2.7 Tape Labels, Dump Labels e EOF Markers

Ci sono due *tape labels* associate con un nastro di backup: la interna *magnetic label* e la esterna *paper label*. La magnetic label è una parte integrante del nastro stesso. Essa contiene:

- Il nome del nastro indicante il suo contenuto. Ci sono tre entry per questo campo:
  - ✓ il nome completo del nastro.
  - ✓ un indicatore di "" o NULL, che può essere creato quando il comando **backup labeltape** è usato senza specificare il nome del nastro. Il sistema di Backup, al momento del Dump, sostituisce l'indicazione vuota con il nome corretto.
  - ✓ nessun nome indica che il nastro non è usato. Il Sistema di Backup genera il nome corretto quando trasferisce i dati sul nastro.
- la dimensione o capacità del nastro. Una lettera indica l'unità di misura (k o K per kilobytes, m o M per megabytes, g o G per gigabytes).

La paper label di un nastro deve contenere le informazioni utili per permettere facilmente l'identificazione per il salvataggio e il recupero dei dati. Queste informazioni sono: il nome del nastro, la ID iniziale del dump, la data del salvataggio e la data di scadenza.

Prima di eseguire sia un dump iniziale o aggiungere un dump su un nastro, il Sistema di Backup crea un *dump level* sul nastro. (Il *dump level* sul primo nastro in un *dump set* rappresenta anche la label del nastro).

Il *dump level* contiene le seguenti informazioni:

- Il nome del nastro contenente il dump.
- La data e l'ora della creazione del dump.
- La cella origine del dump.
- La dimensione del dump in kilobytes.
- Il *dump level* del dump.
- Il dump ID del dump.

Il Sistema di Backup aggiunge un **EOF (end-of-file)marker** per ogni volume che viene salvato sul nastro. L' *EOF marker* può variare da 2 kilobytes a più di 2 megabytes, a seconda del tipo di unità nastro usata e la sua dimensione con la dimensione totale del nastro, deve essere configurata nel Sistema di Backup.

## 2.8 Macchina Tape Coordinator

I salvataggi ed i recuperi sono fisicamente ottenuti tramite una macchina *Tape Coordinator*. Per qualificare come macchina *Tape Coordinator*, una macchina deve avere i seguenti requisiti:

- essere in una sicura locazione.
- essere una macchina client AFS.
- avere uno o più drive nastro attaccati.
- essere configurata come macchina *Tape Coordinator*.
- deve essere attivata come *Tape Coordinator* con il programma **butc**.

Il *Tape Coordinator* deve essere attivo quando si usa il comando **backup**. Una univoca *port offset number* identifica ogni unità nastro.

## 2.9 Il Database del Backup

Una grande quantità di dati usata dal Sistema di Backup è memorizzata nel *Backup Database*. Esiste solo un *Database di Backup* per cella e comunque una copia è memorizzata nel *database del server*.

Il Database di Backup contiene le seguenti informazioni:

- Il **nome del host** e il **numero della port offset**.
- I **livelli del dump** associati con la gerarchia del dump e la **data di scadenza** associata con ogni **livello del dump**.
- Il **volume set** che può essere salvato e le **volume entries** che compongono ogni *volume set*.

- Un record per ogni **dump**, il nome di ogni nastro contenente quel dump, ogni volume contenuto in quel dump, il livello del dump, la data di scadenza, il dump ID del dump iniziale ed infine se è un dump aggiunto o iniziale.

### 3. Configurazione del Sistema di Backup

Prima che il Sistema di Backup possa essere usato per il salvataggio e il recupero dei dati devono essere eseguite le seguenti operazioni:

- Determinare la dimensione dei nastri e la dimensione del *EOFMarker* per ogni unità nastro.
- Configurare la macchina che funzionerà da *Tape Coordinator*.
- Definire i *volume set* ed i *volume entries*.
- Configurare i livelli di dump per creare la gerarchia del dump.
- Inizializzare i nastri.

#### 3.1 Determinazione della capacità del nastro e del *EOF Marker*

La dimensione di un nastro determina la somma dei dati che il Sistema di Backup può mettere su di esso e varia per nastri e unità nastro differenti. In aggiunta, il Sistema di Backup aggiunge un *end of file* (EOF) *marker* per ogni volume che viene salvato su nastro. La dimensione del *marker* riguarda la somma di spazio disponibile per i dati sul nastro. Questi valori sono specificati nel file */usr/afs/backup/tapeconfig* per ogni unità nastro. Con il comando **fms** (*file-mark-size*) si determina la dimensione del nastro o la dimensione del *EOF marker*.

==> Per determinare la capacità del nastro e la dimensione del *EOF marker* è necessario svolgere i seguenti passi:

**Passo 1:** Verificare che esista il file *fms.log*, in caso contrario verificare che si possa scrivere nella directory corrente.

**Passo 2:** Inserire un nastro nell'unità. Poiché il comando *fms* sovrascrive il nastro è bene usare un nastro vuoto.

**Passo 3:** Digitare il comando **fms** :

```
% fms <tape special file>
dove:
```

**fms** deve essere scritto per intero.

*tape special file* specifica il nome del device Unix dell'unità nastro per il quale si vuol determinare la dimensione del nastro e del *EOF marker*.

Un esempio del comando *fms* è il seguente:

```
% fms /dev/rmt0h
```

```
wrote block: 130408
Finished data capacity test – rewinding
Wrote 1109 blocks, 1109 file marks
Finished file mark test
Tape capacity is 2136604672 bytes
Files marks are 1910205 bytes
```

### 3.2 Configurazione della macchina *Tape Coordinator*

⇒ Per determinare la capacità del nastro e la dimensione del *EOF marker* è necessario svolgere i seguenti passi:

**Passo 1:** Installare una o più unità nastro sulla macchina che funzionerà da *Tape Coordinator*.

**Passo 2:** Verificare che l'utente preposto alla configurazione sia incluso nel file */usr/afs/etc/UserList*.

**Passo 3:** Verificare che i file binari **backup**, **butc** e **fms** siano nella directory */usr/afsws/etc*.

**Passo 4:** Diventare "root" nella macchina Unix scelta come *Tape Coordinator*.

**Passo 5:** Creare la directory */usr/afs/backup* nella macchina *Tape Coordinator*.

**Passo 6:** Creare il file */usr/afs/backup/tapeconfig* con un *text editor* includendo una singola linea nel file per ogni unità nastro attaccata. Ogni linea deve contenere i seguenti elementi separati da spazio:

- **La dimensione di default dei nastri** segue le seguenti abbreviazioni:

- Kilobytes: k o K (per esempio, 2k o 2K)
- Megabytes: m o M (per esempio, 2m o 2M)
- Gigabytes: g o G (per esempio, 2g o 2G)

- **La dimensione del end-of-file (EOF) marker**

- **Il nome del device UNIX** dell'unità nastro (per esempio */dev/rmt1h*).

- Il numero di **port offset** del *Tape Coordinator* associato con l'unità nastro. Poiché il Sistema di Backup può gestire sino ad un massimo di 58,511 drive per cella, i valori per la *port offset* vanno da 0 ad 585510. Il default della *port offset* per i comandi di Backup è 0. Il seguente è un esempio del file *tapeconfig* per una macchina che ha due unità nastro. I rispettivi numeri di *port offset* per i due driver sono 0 e 1.

```
2g 1m /dev/rmt0h 0
2G 1M /dev/rmt1h 1
```

**Passo 7:** Decidere a quale utente e gruppo debbono appartenere la directory */usr/afs/backup* ed il file */usr/afs/backup/tapeconfig*. Poiché il *Tape Coordinator* crea e scrive il file di log ed il file degli errori nella directory */usr/afs/backup*, l'utente per attivare il *Tape Coordinator* deve avere il permesso di accedere in lettura e scrittura tale directory.

**Passo 8:** Digitare il comando **backup addhost** per creare una entry nel *Database di Backup*.

---

```
% backup addhost <tape machine name> [<TC port offset>]
dove:
```

**addh** è l'abbreviazione di **addhost**.

*tape machine name* è l'indirizzo internet completo della macchina *Tape Coordinator*.

*TC port offset* è il numero di *port offset*.

### 3.3 Aggiunta o Rimozione di una macchina Tape Coordinator

Un *Tape Coordinator* è associato ad ogni unità nastro. Poichè ad ogni *Tape Coordinator* deve essere assegnato un numero di *port offset*, tutti i comandi **backup** che usano una unità nastro hanno come argomento **-portoffset**. La seguente sezione spiega i passi che necessitano per aggiungere o rimuovere una macchina *Tape Coordinator*.

==> Per aggiungere un *Tape Coordinator* è necessario svolgere i seguenti passi:

**Passo 1:** Installare l'unità nastro sulla macchina che deve funzionare come *Tape Coordinator*.

**Passo 2:** Verificare che l'utente preposto a tale operazione sia incluso nel file */usr/afs/etc/UserList*.

**Passo 3:** Scegliere il numero di *port offset* per l'unità nastro. Usare il comando **backup listhosts** per testare se ci sono assegnamenti precedenti di *port offset*.

**Passo 4:** Usando un qualsiasi editore, aggiungere una linea di definizione del nuovo *Tape Coordinator* nel file *tapeconfig*. Questa linea di definizione deve contenere:

- **La dimensione di default dei nastri**
- **La dimensione del *end-of-file(EOF)* marker**
- **Il nome del device UNIX dell'unità nastro (per esempio */dev/rmt1h*).**
- **Il numero di port offset del *Tape Coordinator* associato con l'unità nastro.** Poiché il Sistema di Backup può gestire sino ad un massimo di 58,511 drives per cella, i valori per la *port offset* vanno da 0 ad 585510. Il default della *port offset* per i comandi di Backup è 0.

**Passo 5:** Digitare il comando **backup addhost** per creare una entry nel database di Backup.

```
% backup addhost <tape machine name> [<TC port offset>]
dove:
```

**addh** è l'abbreviazione di **addhost**

==> Per rimuovere un *Tape Coordinator* è necessario svolgere i seguenti passi:

**Passo 1:** Usando un qualsiasi editore, rimuovere la definizione del Tape Coordinator dal file */usr/afs/backup/tapeconfig*.

**Passo 2:** Digitare il comando **backup delhost** per cancellare l'entry dal *Database di Backup*.

```
% backup delhost <tape machine name> [<TC port offset>]
dove:
```

**delh** è l'abbreviazione **di delhost**

### 3.4 Definizione dei *Volume Sets* e *Volume Entries*

Prima di fare qualsiasi backup è necessario creare uno o più *volume set* nel *Database di Backup*. Un *volume set* definisce un gruppo di uno o più volumi che possono esser salvati. Una volta definito un *volume set* è necessario definire le entry in quel set. Ogni entry di un volume definisce il volume e la locazione del file server di una macchina. Ogni volume entry consiste di tre campi ed ogni campo è separato da spazio:

- **file server machine name:** è il nome completo del *file-server* che ospita il volume. È possibile specificare una singola macchina *file-server* o usare il metacarattere \* per specificare tutti i *file-server* che sono nel VLDB.
- **partition name:** è il nome completo (per esempio, */vicepx*) della partizione su cui risiede il volume. È possibile specificare una singola partizione o usare il metacarattere \* per specificare tutte le partizioni presenti nel VLDB.
- **volume name:** è il nome del volume che deve essere salvato. È possibile usare qualsiasi combinazione di caratteri e metacaratteri per definire il nome del volume.

⇒ Per aggiungere un volume set al Backup Database è necessario eseguire il seguente passo:

**Passo 1:** Eseguire il comando **backup addvolset**, Il *volume set* rimane vuoto fino a che viene usato il comando **backup addvolentry** per definire in esso le entry.

```
% backup addvolset <volume set name>
```

dove:

**addvols** è l'abbreviazione di **addvolset**.

*volume set name* è il nome del *volume set* che deve essere aggiunto.

⇒ Per aggiungere un *volume entry* ad un *volume set* è necessario eseguire il seguente passo:

**Passo 1:** Eseguire il comando **backup addvolentry** per definire i *volume entry* del *volume set* precedentemente creato con il comando **backup addvolset**.

```
% backup addvolentry <volume set name> <machine name>  
                        <partition name> <volume name>
```

dove:

**addvole** è l'abbreviazione di **addvolentry**.

*partition name* è il nome della partizione che ospita il volume. L'espressione metacarattere *.\** può essere usata per definire tutte le partizioni.

*volume name* specifica il volume che deve essere incluso nel *volume entry*. L'espressione *.\** può essere usata per specificare i nomi di più volumi.

⇒ Per cancellare un *volume set* dal *Database di Backup* è necessario eseguire il seguente passo:

**Passo 1:** Eseguire il comando **backup delvolset** per cancellare ogni *volume set*. Non è necessario cancellare prima le entry in ogni *volume set* perchè esse sono cancellate automaticamente.

% **backup delvolset** <*volume set name*>  
dove:

**delvols** è l'abbreviazione di **delvolset**.

⇒ Per cancellare un *volume entry* da un *volume set* è necessario eseguire i seguenti passi:

**Passo 1:** Eseguire il comando **backup listvolsets** per determinare il numero indice dell'entry che vogliamo cancellare. Questo è necessario solo se il *volume set* contiene più entry. Se il set contiene una singola entry, l'indice è 1.

% **backup listvolsets** <*volume set name*>  
dove:

**listv** è l'abbreviazione di **listvolsets**.

**Passo 2:** Eseguire il comando **backup delvolentry** per cancellare la *volume entry*.

% **backup delvolentry** <*volume set name*> <*volume entry index*>  
dove:

**delvole** è l'abbreviazione di **delvolentry**.

*volume entry index* è il numero indice dell'entry che deve essere cancellata.

### 3.5 Creazione della gerarchia del dump

La gerarchia di un dump è definita dai livelli del dump che lo compongono. Quando si definisce un *dump level* è necessario seguire le seguenti regole:

- Il nome di un *dump level* può avere un qualsiasi numero di livelli. Il segno “/” viene usato per separare i differenti livelli nella gerarchia del dump.
- Il nome del *dump level* non deve contenere il punto. Quando un dump viene trasferito sul nastro, il nome del *dump level* diventa parte del nome del dump e gli elementi del nome del dump (il nome del *volume set* e l’ultimo elemento del nome del *dump level*) sono aggiunti con un punto.
- Qualsiasi carattere all’infuori del punto può formare il nome del *dump level*.
- La massima lunghezza per un qualsiasi livello è 28 caratteri.
- La massima lunghezza per il nome di un *dump level*, *full* o *incremental*, è di 256 caratteri. Esso include qualsiasi slash (/) che sono parte del nome.

### 3.6 Definizione della data di scadenza per i *Dump Levels*

Ogni *dump level* è associato con una data di scadenza che è inizialmente definita con il comando **backup adddump** e può essere cambiata con il comando **backup setexp**. Entrambi comandi hanno l’opzione **-expires** per impostare la data di scadenza. Le date di scadenza possono essere definite in modo relativo e modo assoluto:

- **Absolute expiration dates** usa la chiave “**at**” per rappresentare una specifica data.

[**at**] <mm/dd/yy> [,hh:mm>] per esempio: *at 11/22/94 18:00*

o

[**at**] NEVER significa che il dump non scadrà mai

- **Relative expiration dates** usa la chiave “**in**” per indicare un numero di anni, mesi e giorni

[**in**] [<integer>.y][<integer> m][<integer>d] per esempio: *in 1y6m2d*

Si può omettere l’argomento **-expires** per specificare che nessuna data di scadenza è associata con un *dump level*.

==> Per aggiungere una data di scadenza ad un *dump level* è necessario eseguire i seguenti passi:

**Passo 1:** Eseguire il comando **backup** per entrare in modo interattivo.

**Passo 2:** Eseguire il comando **backup adddump** per definire uno o più *dump level*.

```
% backup adddump -dump <dump level name>
```

```
[-expires <expiration date>]
```

dove:

**addd** è l’abbreviazione di **adddump**.

|                        |  |
|------------------------|--|
| <i>dump level name</i> | identifica ogni nuovo <i>dump level</i> che deve essere aggiunto. Se si specifica più di un <i>dump level</i> deve essere usato lo switch <b>-dump</b> . È necessario fornire l'intero <i>pathname</i> del <i>dump level</i> e precedere ogni livello nel <i>pathname</i> con uno slash (/). |
| <i>expiration date</i> | imposta la data di scadenza con ogni <i>dump level name</i> .  |

==> Per cambiare la data di scadenza di un *dump level* è necessario eseguire il seguente comando:

```
% backup setexp -dump <dump level name>
[-expires <expiration date>]
dove:
```

**se** è l'abbreviazione di **setexp**.

==> Per cancellare un *dump level* è necessario eseguire il seguente comando:

```
% backup deldump <dump level name>
dove:
```

**deld** è l'abbreviazione di **deldump**.

### 3.7 Inizializzazione di un nastro

La label magnetica di un nastro può contenere:

- Il nome completo del nastro
- Un indicatore di "" o NULL che può essere creato quando si usa il comando **backup labeltape** senza specificare il nome del nastro. Il Sistema di Backup sovrascrive l'indicatore vuoto con il nome corretto del dump.
- Nessuna label, indica che il nastro non è usato. Il Sistema di Backup genera il nome corretto quando trasferisce un dump sul nastro.

Non è essenziale inizializzare i nastri prima del trasferimento dei dati, poiché il Sistema di Backup può usare nastri non inizializzati, parzialmente inizializzati o totalmente inizializzati. È necessario inizializzare di nuovo un nastro se:

- Si deve creare un nuovo *dump set* sul nastro e il nome dell'iniziale dump per il nuovo *dump set* è diverso dal nome dell'iniziale dump del corrente *dump set*. Il Sistema di Backup infatti non permette la creazione di un nuovo *dump set* contenente un *dump set* precedente se i nomi dell'iniziale dump non sono uguali.
- Si vuole riusare un nastro prima che la data di scadenza associata con il dump sia trascorsa. Il Sistema di Backup non permette il riciclo di un nastro che contiene un *dump set* con uno o più dump non scaduti.

- Se la capacità del nastro è differente dalla capacità dell'unità nastro che si vuole usare. La capacità di un nastro per un'unità nastro è memorizzata nel file */usr/afs/backup/tapeconfig*

==> Per inizializzare un nastro è necessario eseguire i seguenti passi:

**Passo 1:** Attivare il Tape Coordinatore.

**Passo 2:** Digitare il comando **backup labeltape**.

```
% backup labeltape [<tape name, defaults to NULL>]
[-size <tape size in Kbytes, defaults to size in tapeconfig>]
[-portoffset <TC port offset>]
dove:
```

**la** è l'abbreviazione di **labeltape**.

*tape name, defaults to NULL*

è il nome nuovo del nastro. Esso deve avere la forma *<volume setname.<dump levelname>.<tape index>*. Omettere questa opzione, significa che la label del nastro è vuota e avrà l'indicatore NULL.

*tape size in kbytes, defaults to size in tapeconfig*

è la capacità del nastro. Va specificata una dimensione solo se la dimensione del nastro è differente da quella memorizzata nel file */usr/afs/backup/tapeconfig*. Se si omette questo argomento, la dimensione del file *tapeconfig* è registrata nella label come la dimensione del nastro.

==> Per leggere una label di un nastro è necessario eseguire il seguente comando:

```
% backup readlabel [<TC port offset>]
dove:
```

**rea** è l'abbreviazione di **readlabel**.

Il seguente esempio legge la label del nastro nell'unità con *port offset* 1

```
% backup readlabel 1
```

```
Tape read was labelled: quests.monthly.1 (697065340)
Size:2150000KBytes
```

L'esempio visualizza nella finestra del Tape Coordinatore la seguente informazione:

Tape label

```
-----  
name =          guetss.monthly.1  
creationTime =  Fri Feb 7 05:31:32 1992  
cell =          transarc.com  
size =          2150000 kbytes  
2 dump path =  /monthly  
dump id =       697065340  
useCount =      44  
-- End of tape label --
```

#### 4. L'interfaccia con il Sistema di Backup

Per eseguire i comandi che interagiscono con un *Tape Coordinator*, devono essere aperte due finestre, una finestra per il *Tape Coordinator* ed una finestra per la *shell* di comandi. I comandi di backup possono essere eseguiti sia in modo *interattivo* che in modo *batch*. I vantaggi di usare il modo interattivo sono:

- La possibilità di visualizzare le operazioni correnti e pendenti con il comando **jobs** che non può essere usato in modo *batch*.
- La possibilità di cancellare le operazioni correnti e pendenti con il comando **kill** che non può essere usato in modo *batch*.
- Non è necessario racchiudere i metacaratteri tra i doppi apici.

Gli *Idnumber* associati alla sessione interattiva sono:

- **jobID** che è un numero assegnato per ogni operazione di backup iniziata. Tale numero è temporaneo ed è usato per tracciare e cancellare le operazioni correnti e pendenti.
- **taskID** che viene assegnato alle operazioni sul nastro. Il *Tape Coordinator* ha il proprio task ID che viene registrato nel file di log e nel file di errore. Per vedere il task ID associato con il *Tape Coordinator* può essere usato il comando **backup status**.

==> Per entrare in modo interattivo è necessario eseguire il seguente passo:

**Passo 1:** Digitare il comando **backup**. Il prompt del modo interattivo appare:

```
% backup
backup>
```

==> Per uscire dal modo interattivo è necessario eseguire il seguente passo:

**Passo 1:** Digitare *quit* per uscire dal modo interattivo

```
Backup > quit
```

Si può uscire dal modo interattivo solo se tutti i jobs nella sessione sono terminati. Durante la sessione interattiva viene stabilita una temporanea connessione con il *Backup Server* e il *VI Server*.

==> Per cancellare le operazioni in modo interattivo eseguire il comando:

```
backup>jobs
dove j è l'abbreviazione di jobs
```

Le seguenti informazioni sono visualizzate per ogni operazione pendente od in esecuzione:

- Il job Id number.
- Uno dei seguenti nomi descrittivi l'operazione:
  - *Dump* per una operazione iniziata con il comando **backup dump**.
  - *Restore* per una operazione con il comando **backup volrestore** o **backup diskrestore**.
  - *Labeltape* per una operazione di inizializzazione di un nastro iniziata con il comando **backup labeltape**.
  - *Scantape* per una operazione iniziata di **backup scantape**.
  - *SaveDb* per una operazione di **backup savedb**.
  - *RestoreDb* per una operazione di **backup restoredb**.
- Il numero dei kilobytes trasferiti dal *file system* al nastro e viceversa per le operazioni di dump e di restore
- La stringa *volume* seguita dal nome che viene salvato per operazioni di dump. Per operazioni di restore, la stringa *restore.volume*
- Uno dei seguenti messaggi indicanti lo stato dell'operazione, (nessun messaggio è visualizzato se l'operazione è eseguita normalmente):
  - [*abort request*] significa che è stato usato il comando di *backup kill*, ma non è stato inviato al *Tape Coordinator*.
  - [*abort sent*] significa che l'operazione è stata cancellata.
  - [*butc contact lost*] significa che il comando backup non può accedere al *Tape Coordinator*.
  - [*drive wait*] significa che l'operazione è in attesa che la specificata unità nastro si liberi.
  - [*operator wait*] significa che l'operatore deve inserire il nastro nell'unità.

**Passo 2:** digitare il comando di *backup kill* per cancellare l'operazione

```
backup> kill <job ID or dump set name >
dove:
```

**k** è l'abbreviazione del comando **kill**

*job Id* or *dump set name*

rappresenta il numero *job Id* o il nome del dump creato nella forma *<volume set name>.<dump level name>*

## 4.1 La finestra del *Tape Coordinator*

Prima di ottenere una operazione che richiede una unità nastro (come dump o restore ) è necessario attivare il *Tape Coordinator* affinché esegua l'operazione.

⇒ Per attivare il *Tape Coordinator* è necessario eseguire i seguenti passi:

**Passo 1:** verificare che l'utente preposto a tale operazione sia incluso nel file */usr/afs/etc/UserList*.

**Passo 2:** verificare che si possa scrivere nel file di log del *Tape Coordinator* (*TL\_device\_name*) e nel file di error (*TE\_device\_name*) nella directory */usr/afs/backup*.

**Passo 3:** aprire una finestra (*shell*) nella macchina che funziona da *Tape Coordinator*.

**Passo 4:** eseguire il comando **butc** per attivare il *Tape Coordinator*. Tale finestra deve rimanere aperta mentre il *Tape Coordinator* è attivo.

```
% butc [<port offset>] [-debuglevel <trace level>] [-cell<cellname.>] [-aixscsi]
  [-noautoquery]
```

dove:

**butc** deve essere digitato per intero.

*trace level* specifica il tipo di traccia che il *Tape Coordinator* deve produrre sullo standard output. Ci sono due validi valori:

- 0 indica che il *Tape Coordinator* chiede solo l'inserimento di nuovi nastri.
- 1 indica che il *Tape Coordinator* oltre a chiedere l'inserimento di nuovi nastri, visualizza le principali attività che esegue (per esempio i volumi che accede e che salva).

*cellname* specifica la cella su cui è attivo il *Tape Coordinator*.

**-aixscsi** deve esser usato solo se il *Tape Coordinator* ha in esecuzione un sistema operativo AIX 2.2.1.

**-noautoquery** disabilita la richiesta dell'inserimento del nastro.

⇒ Per fermare il *Tape Coordinator* è sufficiente digitare un segnale di *interrupt* come **^c**

---

==> Per visualizzare lo stato del *Tape Coordinator* è necessario eseguire il seguente comando:

**% backup status** [*<TC port offset>*]

dove:

**st** è l'abbreviazione di **status**

Se il *Tape Coordinator* non sta eseguendo nessuna operazione, si avrà il seguente messaggio:

*Tape Coordinator is idle*

In caso contrario si avrà

*Unable to connect to tape coordinator at port <TC port offset>.*

Altrimenti si avranno in risposta le seguenti informazioni:

- Il numero di task ID associato con l'operazione.
- Un nome descrivente l'operazione che il *Tape Coordinator* sta eseguendo.
- Il numero di kilobytes trasferiti dal file-system al nastro (operazione di salvataggio) o dal nastro al file-system ( per recupero delle informazioni).
- La stringa *volume* seguita dal nome del volume che si sta salvando o la stringa *restore.volume*.

## 5. Visualizzazione del Backup

I seguenti comandi possono essere usati per visualizzare il contenuto del *Database di Backup*:

- **backup dbverify** visualizza lo stato del Backup.
- **backup listvolsets** visualizza i *volume set* e le loro entry.
- **backup listdumps** visualizza la gerarchia del dump.
- **backup dumpinfo** visualizza l'informazione sullo stato dei dump.
- **backup listhosts** visualizza il *Tape Coordinator* e i loro numeri di *port offset*.
- **backup volinfo** visualizza la storia del salvataggio di un volume.

==> Per verificare lo stato del *Database di Backup* eseguire il comando:

```
% backup dbverify [-detail]
```

dove:

**db** è l'abbreviazione di **dbverify**.

**-detail** specifica che informazioni aggiuntive sullo stato del *Database di Backup* devono essere visualizzate.

Se il database non è danneggiato si avrà in risposta *Database OK*

==> Per listare i *volume set* e *volume entries* eseguire il comando:

```
% backup listvolsets [<volume set name>]
```

dove:

**listv** è l'abbreviazione di **listvolsets**.

*volume set name* è il nome del *volume set* di cui vogliamo visualizzare le entry. Omettere questo argomento significa che verranno visualizzati tutti i *volume set* e le loro entry definite nel Database.

Per esempio:

```
% backup listv
```

Volume set user:

```
Entry 1: server.*, partition.*, volume:
user.*\backup
```

Volume set sun:

```
Entry 1:server.*, partition.*, volumes: sun3_411\.*
Entry 2:server.*, partition.*, volumes: sun4c_411\.*
```

Volume set rt:

```
Entry 1:server.*, partition.*, volumes: rt_aos4\.*
```

==> Per visualizzare la gerarchia del dump è necessario eseguire il comando:

```
% backup listdumps
dove: listd è l'abbreviazione di listdumps
```

Per esempio:

```
% backup listdumps
/week1 expires in 27d
    /tuesday expires in 13d
        /thursday expires in 13d
/sunday expires in 13d
    /tuesday expires in 13d
        /tuesday expires in 13d
```

==> Per visualizzare l'informazione sullo stato del backup è necessario eseguire il comando:

```
% backup dumpinfo [<no.of dumps>] [-id<dump id>] [-verbose]
dove:
```

**dumpi** è l'abbreviazione di **dumpinfo**.

*no.of dumps*

specifica il numero del dump di cui vogliamo l'informazione. Se si omette questo argomento sarà visualizzata l'informazione degli ultimi 10 dumps.

**-verbose** fornisce ulteriori informazioni sul dump specificato con *dump id*.

Se viene specificato *no.of dumps* sono fornite le seguenti informazioni:

- *dumpid*: specifica il numero ID dello specificato dump di cui vogliamo l'informazione.
- *parentid*: è il numero ID del dump padre. Un valore 0 indica che è un dump completo e quindi non ha riferimenti ad altri dump.
- *lv*: è la profondità della gerarchia del dump. Un valore di 0 significa un dump completo e in questo caso *parentid* è 0. Un valore di 1 o più grande indica un dump incrementale.
- *created*: indica l'ora e il giorno della creazione del dump.
- *nt*: è il numero dei nastri richiesto per il dump. Un valore uguale a 0 sta a significare che l'operazione di dump è fallita ma esiste sempre l'entry nel Database.
- *nvols*: è il numero dei volumi che compongono il dump.
- *dump name*: è il nome del dump nella forma:

---

```
<volume set name>.<dump level name>(initial dump ID)
```

se **-id** è specificato vengono visualizzate le seguenti informazioni:

- La prima linea, con nome *dump* visualizza le seguenti informazioni circa lo specificato nastro:
  - il numero del dump ID.
  - la profondità della gerarchia del *dump level* usata per creare il dump.
  - il numero dei volumi nel dump.
  - la data di creazione del dump.
- La seconda linea, con nome *tape* visualizza il nome del nastro contenente il dump seguito dalla ID del dump iniziale.
- La terza linea visualizza il numero dei volumi e la data di creazione del dump.
- La quarta linea contiene:
  - *Pos:* rappresenta la posizione di ogni volume sul nastro iniziando dalla posizione 2 (la posizione 1 è occupata dalla label del nastro).
  - *Clone time:* per i volumi *Backup* o *ReadOnly* rappresenta la data della copia dalla versione *ReadWrite*.
  - *Nbytes:* la dimensione in bytes del volume.
  - *Volume:* il nome del volume.

Se **-id** e **--verbose** sono specificati, il comando visualizzerà le seguenti informazioni:

- L'intestazione *dump* seguita dal numero di linee che visualizzano informazioni dettagliate sul dump specificato.
- L'intestazione *tape* seguita dal numero di linee che visualizzano informazioni dettagliate sul nastro contenente lo specificato dump.
- Zero o più occorrenze dell'intestazione *volume*, ognuno delle quali è seguita da informazioni dettagliate sui volumi inclusi del dump specificato.

Il seguente esempio visualizza l'informazione degli ultimi 5 *dump set* registrati nel Database:

```
% backup dumpi 5
```

```
dumpid      parentid lv created          nt  nvols  dump name
690169399          0 0 02/02/92 04:12 1 22  usr.sun (690169399)
690428867 690169399 1 02/05/92 05:29 1 62  usr.wed (690169399)
690514342 690169399 1 02/06/92 05:21 1 46  usr.thu (690169399)
690600291 690169399 1 02/07/92 05:23 1 58  usr.fri (690169399)
690874954          0 0 02/09/92 05:31 2 73  sys.week
```

==> Per visualizzare lo stato del *Tape Coordinator* eseguire il comando:

**% backup listhots**  
dove **listh** è l'abbreviazione di **listhosts**

Per esempio:

**% backup listhosts**  
*Tape hosts:*  
*Host backup1.transarc.com, port offset 0*  
*Host backup1.transarc.cm, port offset 2*

==> Per visualizzare la storia del dump di un volume eseguire il comando:

**%backup volinfo <volume name>**  
dove:

**voli** è l'abbreviazione di **volinfo**.

*volume name* è il nome del volume di cui vogliamo visualizzare l'informazione. Se è stata salvata la versione di Backup o ReadOnly, piuttosto della versione ReadWrite, il nome del volume avrà l'estensione **.backup** o **.readonly**.

L'output del comando visualizzerà in sei colonne le seguenti informazioni:

- *dumpID*: è il dump ID del *volume name*.
- *lvl*: rappresenta la profondità della gerarchia del dump level in cui *il volume name* è stato salvato. Un valore uguale a 0 rappresenta un dump completo. Un valore uguale ad 1 o più grande significa un dump incrementale.
- *parentid*: rappresenta il dump ID del dump padre. Un valore uguale a 0 significa che è un dump totale, in questo caso *lvl* sarà 0.
- *creation date*: rappresenta la data e l'ora in cui il Sistema ha iniziato l'operazione di salvataggio che ha creato il *dump set*.
- *clone date*: per i volumi *Backup* o *ReadOnly* rappresenta la data della copia dalla versione *ReadWrite*. Per un volume *ReadWrite* questa è la stessa della data di creazione.
- *tape name*: rappresenta il nome del nastro contenente il dump set nella forma *<volume set name>.<dump level name>.<tape index>(dump ID)*, dove *volume set name* è il nome del *volume set* associato con il dump iniziale nel *dump set*; *dump level name* è il nome dell'elemento finale del *dump level* in cui il dump iniziale è stato salvato; *tape index* è la posizione

numerica del nastro nel *dump set* e *dump ID* è l'ID del dump iniziale del *dump set* di cui il nastro fa parte.

Il seguente esempio mostra la storia di un dump:

**%backup volinfo user.smith.backup**

| DumpId    | lvl | parentID  | creation date  | clone date     | tape name              |
|-----------|-----|-----------|----------------|----------------|------------------------|
| 654998415 | 1   | 654925629 | 02/04/92 05:26 | 02/04/92 05:01 | user.tue.1 (654912115) |
| 654912115 | 1   | 654925629 | 02/03/92 05:25 | 02/03/92 05:05 | user.mon.1 (654912115) |
| 654925629 | 0   | 0         | 02/02/92 05:38 | 02/04/92 04:58 | user.sun.1 (654925629) |

==> Per visualizzare il contenuto di un nastro è necessario eseguire i seguenti passi:

**Passo 1:** Se il processo di *butc* è in esecuzione con il flag *-noautoquery*, è necessario inserire il giusto nastro prima di eseguire il comando **backup scantape**.

**Passo 2:** `% backup scantape [-dbadd] [-portoffset <TC port offset>]`  
dove:

**sc** è l'abbreviazione di **scantape**.

**-dbadd** significa che l'informazione dal nastro deve essere aggiunta al database.

**Passo 3:** Mettere il nastro corretto nell'unità nastro e premere **Return** dalla finestra del Tape Coordinatore.

**% backup scantape 2**

Dump label

```
Name =          guetss.monthly.1
CreationTime = Fri Feb 7 05:31:32 1992
Cell =          transarc.com
Size =          2150000 kbytes
Dump path = /monthly
Dumpi id = 697065340
UseCount = 44
--End of dump label --
```

-- volume --

```
volume name: user.guets10.backup  dumpID 697065340  endTime 0
volume ID 1937573829             level 0          clondate Fri Feb 7 05:03:23 1992
dumpSetName: quests.monthly      parent ID 0
```

## 6. Salvataggio dei Dati

Il comando **backup dump** è usato per salvare i dati AFS su nastro. Con il comando si deve specificare il *volume set* che deve essere salvato ed il livello del backup. Se si specifica un dump completo, sono salvati tutti i files.

Quando un *volume set* è salvato sul nastro, il dump risultante può creare un nuovo *dump set* o essere aggiunto ad un *dump set* esistente. Per aggiungere un dump ad un *dump set* esistente è necessario usare il flag **-append** nel comando **backup dump**.

### 6.1 Il processo di Backup

In questa sezione è riportato un esempio del processo di Backup. Il *volume set usersys* è salvato al *dump level /sunday/wednesday*.

1. Il Sistema di backup legge il volume set *usersys* dal *Database di Backup* per conoscere quali entry esso contiene. Il *volume set* e le sue *entry* sono state create con il comando **backup addvolset** e **backup addvolentry**. In questo esempio, le entries in *usersys* sono:

```
.    *.*user.*
.    *.*sys.*
```

2. Il sistema di Backup scandisce il VLDB per verificare i metacaratteri di ogni volume entry e generare una lista completa dei volumi che devono essere inclusi nel dump.
3. Il Sistema di Backup legge la gerarchia del dump nel *Database di Backup* per cercare se */sunday1/wednesday* è un dump incrementale ed in questo caso per determinare da quale precedente livello dipenda. Nell'esempio *sunday1/wednesday* è incrementale e */sunday1* è il suo padre.

|         |                |                |
|---------|----------------|----------------|
| Sunday1 | expires in 27d |                |
|         | /monday        | expires in 13d |
|         | /tuesday       | expires in 13d |
|         | /wednesday     | expires in 13d |
|         | /thursday      | expires in 13d |
|         | /friday        | expires in 13d |
| Sunday2 | expires in 27d |                |
|         | /monday        | expires in 13d |
|         | /tuesday       | expires in 13d |
|         | /wednesday     | expires in 13d |
|         | /thursday      | expires in 13d |
|         | /friday        | expires in 13d |

Se viene specificato */sunday1*, questo sarà un dump completo e quindi il sistema copierà completamente il contenuto di ogni volume in *usersys*. Poichè */sunday1/wednesday* è un livello di dump incrementale, il *dump set* include solo quei files che hanno subito un cambiamento dal salvataggio di *usersys* al momento del dump */sunday1*

4. Poichè */sunday1* è il padre di */sunday1/wednesday*, il Sistema di Backup cerca nel Database di Backup la data e l'ora del dump *usersys* al livello */sunday1*.
5. Se il dump è un dump iniziale appartenente ad un nuovo *dump set* ed il nastro contiene un precedente dump set che è stato sovrascritto, il Sistema di Backup verifica che:
  - ogni dump nel corrente *dump set* sia scaduto. Se una o più data di scadenza associata con quel *dump set* non è scaduta, l'operatore viene avvertito.
  - il nome del dump iniziale nel nuovo *dump set* corrisponda al nome del dump iniziale del vecchio *dump set*.
6. Il Sistema di Backup trasferisce la lista dei volumi che devono essere salvati nell'appropriato *Volume Server*.
7. Il *Tape Coordinator* designato scrive i dati sul nastro con la data di scadenza associata con il dump. Il Sistema di Backup assegna un numero ID al dump e registra il numero nel *Database di Backup*. Se un nastro non riesce a contenere tutti i dati, il Sistema di Backup avviserà l'operatore per aggiungere ulteriori nastri.

## 6.2 Uso dei nastri con l'operazione di Backup

Prima di fare un backup è necessario assicurarsi che i nastri siano abbastanza capienti e che la loro dimensione sia riportata nel file */usr/afs/backup/tapeconfig*. Il Sistema di Backup aggiunge un nastro inizializzato con la somma dei dati listati come la capacità del nastro nella sua label. Aggiunge un nastro non inizializzato con la somma dei dati listati come la capacità dell'unità nastro nel file */usr/afs/backup/tapeconfig*.

Se un nastro non inizializzato è più grande della dimensione del nastro registrato nel file *tapeconfig*, esso non viene aggiunto al momento del backup. Se un nastro non inizializzato è più piccolo della dimensione registrata nel file *tapeconfig*, qualsiasi volume che non entra completamente nel nastro, sarà rimosso dal nastro dopo un dump parziale e sarà salvato nel successivo nastro del *dump set*. Un volume che è più grande dell'attuale capacità del nastro impedisce il salvataggio del volume sul nastro. Per assicurare un uso ottimale dei nastri, è necessario eseguire il comando **fms** per determinare la giusta dimensione di ogni tipo di nastro usato per il dump.

Un volume può non entrare in un singolo nastro anche se la capacità del nastro corrisponde alla dimensione registrata nel file *tapeconfig*. Se il Sistema di Backup raggiunge la fine del nastro mentre sta salvando un volume, mette i dati che rimangono in un altro nastro e il *Database di Backup* automaticamente registra che il volume risiede su più nastri.

Prima di eseguire un salvataggio, è possibile controllare l'effetto del comando semplicemente includendo il flag **-n** nel comando **backup dump** specificando gli opportuni argomenti. In questo modo è possibile stabilire la dimensione del *volume set* prima di eseguire il dump e calcolare quindi il numero corretto dei nastri necessari.

Dopo il completamento dell'operazione di backup che implica la creazione di un nuovo *dump set*, è necessario registrare il dump ID del dump iniziale su carta registrando la label di ogni nastro del *dump set* permettendo in questo modo di sapere più facilmente quali nastri occorrono per recuperare l'informazione dal dump set.

==> Per il salvataggio di un volume set (creazione di un nuovo *dump set*) occorrono eseguire i seguenti passi:

**Passo 1:** attivare il *Tape Coordinator*. Se il processo **bute** sta eseguendo con il flag **-noautoquery** è necessario inserire il giusto nastro prima di eseguire il comando **backup dump**.

**Passo 2:** decidere quali *volume set* e *dump level* vogliamo usare. Se necessario, usare **backup listvolsets** e **backup listdumps** per visualizzare l'informazione su esistenti *volume set* e *dump level*.

**Passo 3:** controllare di avere a disposizione un sufficiente numero di nastri propriamente inizializzati (se necessario, usare il comando **backup readlabel** per controllare le label). È necessario inizializzare di nuovo qualsiasi nastro che fa parte di un *dump set* contenente uno o più dump non scaduti o che contengono un dump iniziale che è differente dal dump iniziale del nuovo *dump set*. Se la label è errata, usare il comando **backup labeltape**.

**Passo 4:** usare il comando **backup dump** per salvare il *volume set* sul nastro

```
% backup dump <volume set name> <dump level name>
[<TC port offset>] [-n]
dove:
```

**dump**           deve essere scritto per intero.

**-n**               specifica che i nomi di tutti i volumi che dovrebbero essere inclusi nella operazione di dump sono visualizzati senza che il salvataggio avvenga.

**Passo 5:** porre il nastro corretto nell'unità nastro e premere **Return** nella finestra del *Tape Coordinator*. Se più di un nastro è richiesto, è necessario rimanere alla console per rispondere alla richiesta dei successivi nastri, in caso di risposta ritardata, periodicamente un suono richiamerà l'attenzione dell'operatore.

### 6.3 Mettere multipli dump in un dump set

Per default, solo un singolo *volume set* ad un particolare *dump level* può essere salvato su un singolo *dump set*. Comunque, usando il flag **-append** con il comando **backup dump** è possibile aggiungere un dump ad un *dump set* che già contiene uno o più altri dump. In questo modo, si velocizza il processo di backup perché sono richiesti meno cambiamenti di nastri, riduce il numero di nastri richiesti per la memorizzazione dei dati ed aumenta l'utilizzazione del nastro. Le condizioni che devono essere soddisfatte prima che un dump possa essere aggiunto ad un dump set sono:

- Il nastro deve contenere un dump iniziale creato con il comando **backup dump** di AFS 3.3.
- Il dump iniziale deve essere registrato nel *Database di Backup*.

---

Se un nastro non contiene un dump iniziale, l'operazione di dump continuerà come un dump iniziale. Se il dump iniziale non esiste nel *Database di Backup*, l'operatore sarà avvisato per mettere un nastro valido nell'unità nastro ed in questo caso può inserire o un nastro corretto nell'unità nastro o terminare il job di dump e listare il nastro nel *Database di Backup*. Per default, un *dump set* inibisce le date di scadenza associate con i dump nel *dump set*, ad esempio aggiungere un dump con nessuna data di scadenza ad un dump che scadrà il 1/1/94 farà in modo che la data di scadenza dell'intero dump sarà 1/1/94.

⇒ per salvare un volume set (aggiungere un dump ad un *dump set* esistente) è necessario eseguire il seguente comando:

```
% backup dump <volume set name> <dump level name>  
[-portoffset <TC port offset> -append
```

## 7. Procedure di automatizzazione del Sistema di Backup

I due principali modi per aumentare il livello di automatizzazione del sistema di backup sono:

- Disabilitare la necessità di risposte non necessarie dell'operatore della finestra *Tape Coordinator*. Questa è raggiunta usando il flag **-noautoquery** quando si esegue il comando **butc**.
- Eseguire il comando **backup dump** ad ore stabilite. Questo è ottenuto specificando un tempo di esecuzione con l'argomento **-at** nel comando **backup dump** o creando un file contenente le istruzioni del comando **backup dump** e specificando quel file con l'argomento **-file**.

### 7.1 Disabilitare la richiesta per l'inserimento del nastro

Quando si esegue un comando backup che richiede un nastro, l'appropriato *Tape Coordinator*, per default, chiede sia l'inserimento del nastro corretto nell'appropriata unità nastro che la pressione del tasto **Return**. Se il *Tape Coordinator* si accorge che l'unità nastro contiene il nastro corretto, immediatamente esegue l'azione richiesta senza l'intervento dell'operatore. Questa scelta non è di default, ma solo se il programma **butc** viene eseguito con il flag **-noautoquery**. Questa possibilità è utile per eseguire salvataggi senza l'intervento dell'operatore, ma se più nastri sono richiesti per completare l'operazione, un operatore deve essere presente per il cambio dei nastri. Inoltre, se il *Tape Coordinator* incontra delle condizioni di errore, esso le scrive nella finestra del *Tape Coordinator* richiedendo quindi l'intervento dell'operatore per il proseguimento dell'operazione. Alcuni delle più comuni condizioni di errore sono:

- No tape in the drive
- Unexpired tape in the drive
- Write-protected tape in the drive

Questa caratteristica riguarda tutti i comandi **backup** che interagiscono con il *Tape Coordinator* che sono:

- **backup diskrestore**
- **backup dump**
- **backup labeltape**
- **backup readlabel**
- **backup restoredb**
- **backup savedb**
- **backup scantape**
- **backup volrestore**

### 7.2 Schedulazione automatica dei Dumps

È possibile far sì che uno o più comandi **backup dump** siano eseguiti in specifici momenti. Questi tipi di dump fanno riferimento ad un **timed** o **scheduled dump** che sono:

- **backup dump** con l'argomento **-at** per l'esecuzione in un momento specifico.
- **backup dump** con l'argomento **-file** per l'esecuzione di uno o più momenti specifici.

Quando un salvataggio viene eseguito i seguenti controlli sono effettuati:

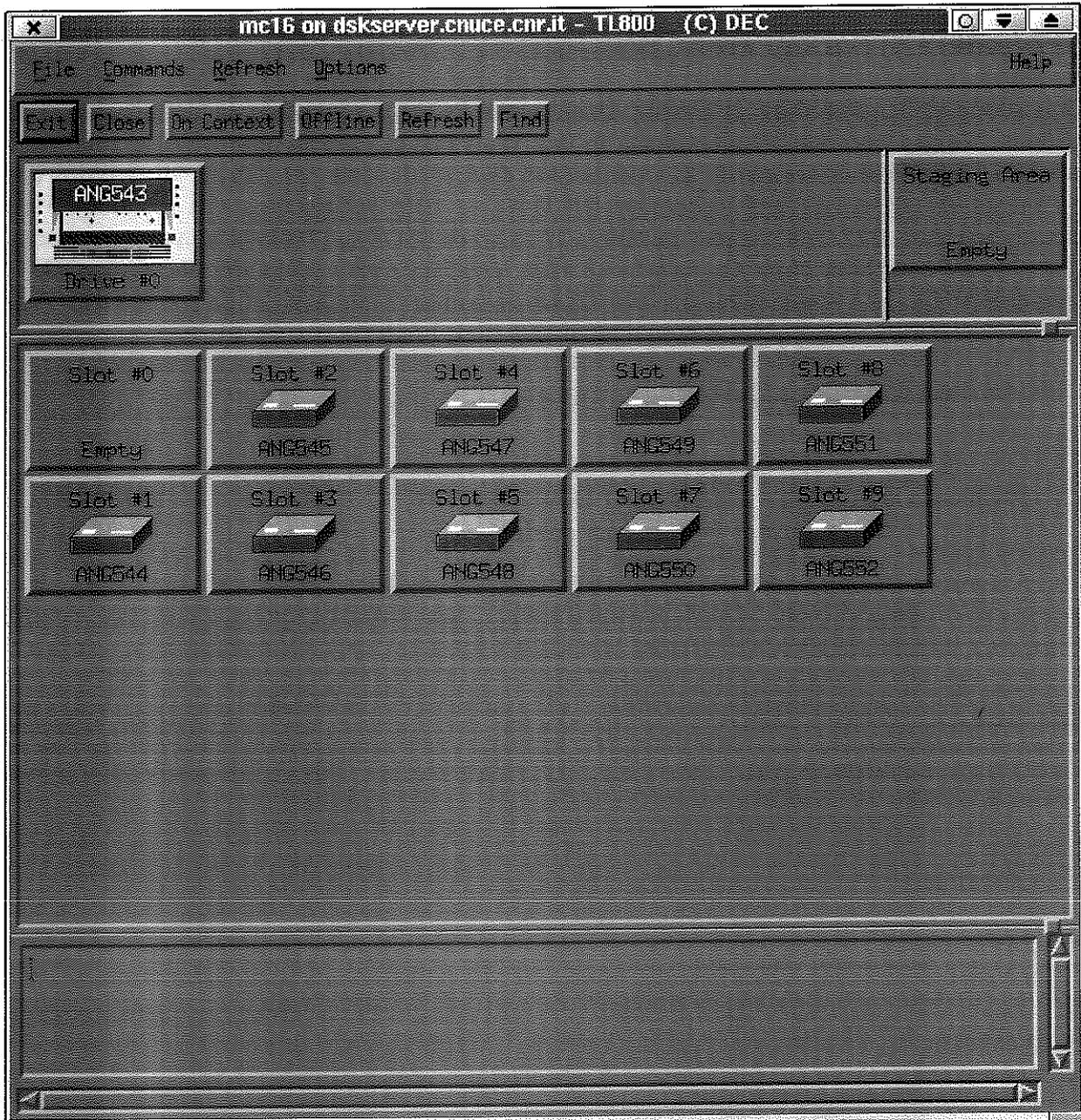


Fig 2: Visualizzazione dell'unità nastro dtl 800

La macchina **dskserver** ed un nodo del SP2 identificato con il nome simbolico **sp08.cnuce.cnr.it** (*Ipaddress:146.48.83.119*) rappresentano i due *file-server* AFS del reparto RCSA ed inoltre sono le due macchine su cui sono definiti tutti gli utenti (AFS) utilizzatori delle risorse del reparto.

Su entrambe le macchine, nel file `/usr/afs/local/BosConfig` in cui sono definiti tutti i processi di esecuzione di AFS, sono stati aggiunte rispettivamente le seguenti entry:

- **dskserver:**

```

bnode cron backvolumes 1
parm /usr/afs/bin/vos backupsys -server dskserver.cnuce.cnr.it -localauth
parm 23:00
end

```

- **sp08:**

```

bnode cron backvolumes 1
parm /usr/afs/bin/vos backupsys -server sp08.cnuce.cnr.it -localauth
parm 23:15
end

```

Le suddette entry permettono per ogni volume definito nei *file-server* **Sp08** e **Dskserver** la creazione di una copia di backup.

Sul *Tape Coordinator*, in questo caso **dskserver** è stato creato il *volume set server-dsks-sp08* e la sua rispettiva gerarchia. Quindi nel nostro caso per ogni versione di backup di ogni volume definito su ogni *file-server* viene effettuato un dump totale il *lunedì* ed una versione di dump incrementale il *mercoledì* ed il *venerdì*.

Volume set server-dsks-sp08:

```
Entry 1: server .*, partition .*, volumes: *.backup
```

```
/lunedì expires in 7d
```

```
/mercoledì expires in 7d
```

```
/venerdì expires in 7d
```

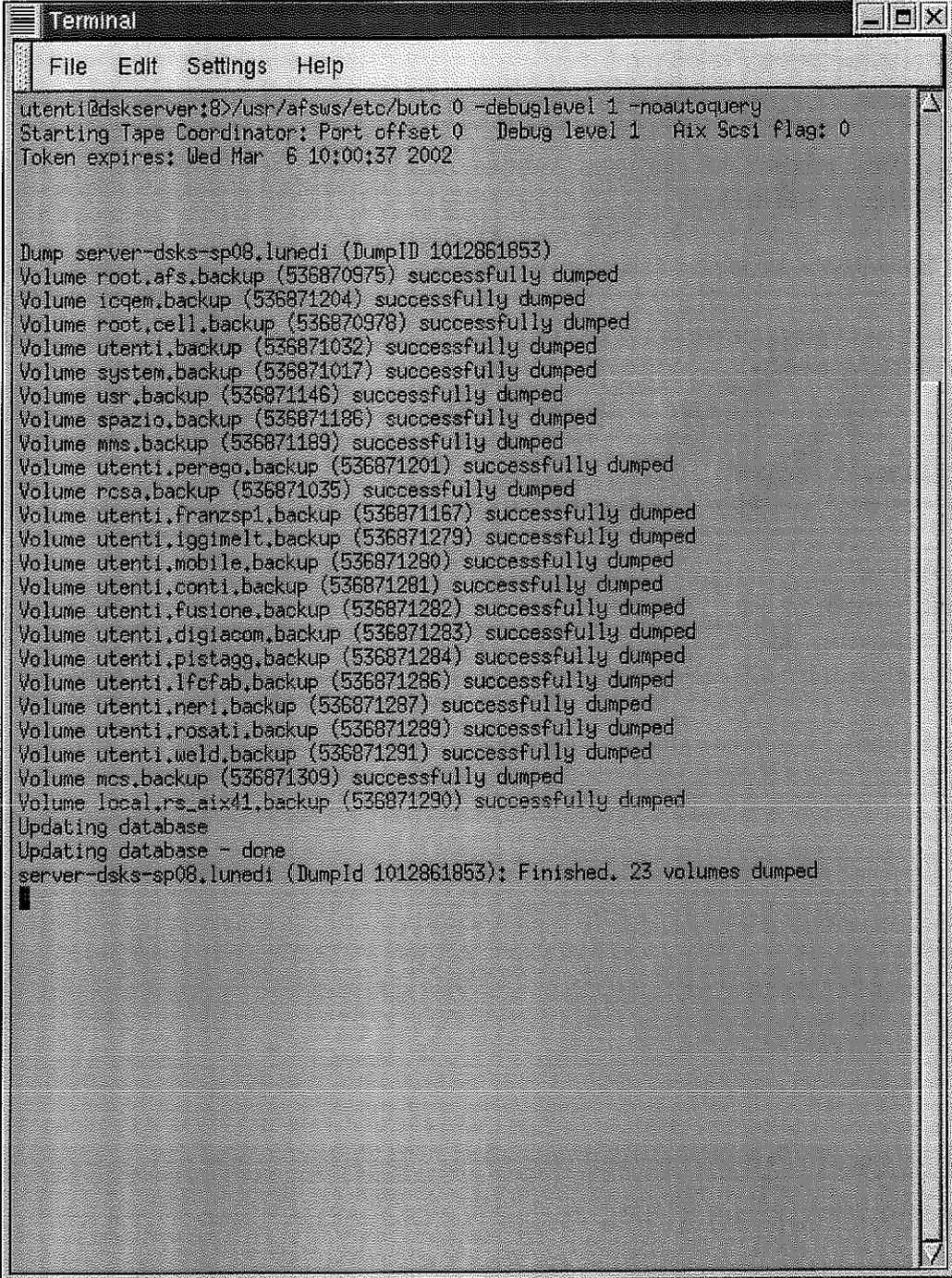
Alle 23:30 di ogni giorno viene attivata la procedura `/afs/cnuce/utenti/salva >> /tmp/Backup 2>&1` la quale in pratica attiva il salvataggio dei dati dei server **dskserver** ed **sp08** sui nastri. (vedi fig. 3 e fig. 4).

```

Terminal
File Edit Settings Help
utente@dskserver:9>backup
backup> dumpi
  dumpid  parentid lv created      nt nvols dump name
1011652254  0 0 01/21/02 23:30  1  23 server-dsks-sp08, lunedì
1012257061  0 0 01/28/02 23:31  1  23 server-dsks-sp08, lunedì
1012429855  0 0 01/30/02 23:30  1  23 server-dsks-sp08, lunedì
1012602654  0 0 02/01/02 23:30  1  23 server-dsks-sp08, lunedì
backup> dumpi
  dumpid  parentid lv created      nt nvols dump name
1012257061  0 0 01/28/02 23:31  1  23 server-dsks-sp08, lunedì
1012429855  0 0 01/30/02 23:30  1  23 server-dsks-sp08, lunedì
1012602654  0 0 02/01/02 23:30  1  23 server-dsks-sp08, lunedì
1012661853  0 0 02/04/02 23:30  1  23 server-dsks-sp08, lunedì
backup> █

```

Fig 3: Visualizzazione dello stato del backup

A terminal window titled "Terminal" with a menu bar containing "File", "Edit", "Settings", and "Help". The terminal shows the execution of a backup command: "utenti@dskserver:8>/usr/afsws/etc/buto 0 -debuglevel 1 -noautoquery". It then displays the output of the backup process, listing 23 volumes and their backup status. The volumes listed are: root.afs.backup, icqem.backup, root.cell.backup, utenti.backup, system.backup, usr.backup, spazio.backup, mms.backup, utenti.perego.backup, rosa.backup, utenti.franzspi.backup, utenti.iggimelt.backup, utenti.mobile.backup, utenti.conti.backup, utenti.fusione.backup, utenti.digiacom.backup, utenti.pistagg.backup, utenti.lfcfab.backup, utenti.neri.backup, utenti.rosati.backup, utenti.weld.backup, mcs.backup, and local.rs\_aix41.backup. The process concludes with "Updating database" and "Updating database - done", followed by the summary: "server-dsks-sp08.lunedì (DumpId 1012861853): Finished. 23 volumes dumped".

```
utenti@dskserver:8>/usr/afsws/etc/buto 0 -debuglevel 1 -noautoquery
Starting Tape Coordinator: Port offset 0  Debug level 1  Aix Scsi flag: 0
Token expires: Wed Mar  6 10:00:37 2002

Dump server-dsks-sp08.lunedì (DumpID 1012861853)
Volume root.afs.backup (536870975) successfully dumped
Volume icqem.backup (536871204) successfully dumped
Volume root.cell.backup (536870978) successfully dumped
Volume utenti.backup (536871032) successfully dumped
Volume system.backup (536871017) successfully dumped
Volume usr.backup (536871146) successfully dumped
Volume spazio.backup (536871186) successfully dumped
Volume mms.backup (536871189) successfully dumped
Volume utenti.perego.backup (536871201) successfully dumped
Volume rosa.backup (536871035) successfully dumped
Volume utenti.franzspi.backup (536871167) successfully dumped
Volume utenti.iggimelt.backup (536871279) successfully dumped
Volume utenti.mobile.backup (536871280) successfully dumped
Volume utenti.conti.backup (536871281) successfully dumped
Volume utenti.fusione.backup (536871282) successfully dumped
Volume utenti.digiacom.backup (536871283) successfully dumped
Volume utenti.pistagg.backup (536871284) successfully dumped
Volume utenti.lfcfab.backup (536871286) successfully dumped
Volume utenti.neri.backup (536871287) successfully dumped
Volume utenti.rosati.backup (536871289) successfully dumped
Volume utenti.weld.backup (536871291) successfully dumped
Volume mcs.backup (536871309) successfully dumped
Volume local.rs_aix41.backup (536871290) successfully dumped
Updating database
Updating database - done
server-dsks-sp08.lunedì (DumpId 1012861853): Finished. 23 volumes dumped
```

Fig 4: Visualizzazione del salvataggio dei volumi degli utenti

## 10. Appendice A

Procedura salva:

```
#!/usr/bin/ksh
#
#
#
mailaddr="giancarlo.bartoli@cnuce.cnr.it"
typeset -R3 giorno
typeset -R5 drive
typeset -R6 nastro
typeset -R1 Plun
typeset -R4 TAPE
typeset -i nerr=0
export MRU_ROBOT=/dev/mc16

#####
# Controllo il giorno. Lunedì, Mercoledì e Venerdì sono i giorni del backup
# e pertanto attivo il Tape Coordinatore
#####

set giorno = `date +%a`
giorno=$3
if [[ $giorno != "Mon" ]] && [[ $giorno != "Wed" ]] && [[ $giorno != "Fri" ]]
then exit
else
set TAPE = `usr/afsws/etc/backup status `
TAPE=$6
if [[ $TAPE != "idle" ]] then
`usr/afsws/etc/butc 0 -debuglevel 1 -noautoquery ` &
fi
fi

#####
Lunedì
#####
if [[ $giorno = "Mon" ]] then
echo "XYZ" >/tmp/zzzxxx
`usr/afsws/bin/klog utenti -pipe < /tmp/zzzxxx`
`rm /tmp/zzzxxx`
set riga = `usr/bin/robot show drive 0 | awk '{print$3}`"
drive=$3
if [[ $drive = "Empty" ]] then
#####
# Se è lunedì, controllo il file data, se contiene 1 è il primo lunedì
#####
cat /afs/cnuce.pi.cnr.it/utenti/bin/data | read Plun
if [[ $Plun = "1" ]] then
`usr/bin/robot load drive 0 slot 6`
`usr/afsws/etc/backup dump server-dsks-sp08 /lunedì 0`
`echo "2" > /afs/cnuce.pi.cnr.it/utenti/bin/data`
```

```
        exit
    else
        `/usr/bin/robot load drive 0 slot 9`
        `/usr/afsws/etc/backup dump server-dsks-sp08 /lunedì 0`
        `echo "1" > /afs/cnuce.pi.cnr.it/utenti/bin/data`
        exit
    fi
else
    set riga = `/usr/bin/robot show drive 0 | awk '{print$3}'`
    nastro=$3
    case $nastro in
        ANG543)
            `/usr/bin/robot unload drive 0 slot 0`
            ;;
        ANG544)
            `/usr/bin/robot unload drive 0 slot 1`
            ;;
        ANG545)
            `/usr/bin/robot unload drive 0 slot 2`
            ;;
        ANG546)
            `/usr/bin/robot unload drive 0 slot 3`
            ;;
        ANG547)
            `/usr/bin/robot unload drive 0 slot 4`
            ;;
        ANG548)
            `/usr/bin/robot unload drive 0 slot 5`
            ;;
        ANG549)
            `/usr/bin/robot unload drive 0 slot 6`
            ;;
        ANG550)
            `/usr/bin/robot unload drive 0 slot 7`
            ;;
        ANG551)
            `/usr/bin/robot unload drive 0 slot 8`
            ;;
        ANG552)
            `/usr/bin/robot unload drive 0 slot 9`
            ;;
    esac
    set riga = `/usr/bin/robot show drive 0 | awk '{print$3}'`
    drive=$3
    if [[ $drive = "Empty" ]] then
        cat /afs/cnuce.pi.cnr.it/utenti/bin/data | read Plun
        if [[ $Plun = "1" ]] then
            `/usr/bin/robot load drive 0 slot 6`
            echo "XYZ" >/tmp/zzzxxx
            `/usr/afsws/bin/klog utenti -pipe < /tmp/zzzxxx`
            `rm /tmp/zzzxxx`
        fi
    fi
fi
```

```

        \usr/afsws/etc/backup dump server-dsks-sp08 /lunedì 0`
echo "2" > /afs/cnuce.pi.cnr.it/utenti/bin/data
set riga = \usr/bin/robot show drive 0 | awk '{print$3}'
nastro=$3
date >> /afs/cnuce.pi.cnr.it/utenti/bin/story
echo $nastro >> /afs/cnuce.pi.cnr.it/utenti/bin/story
exit
    else
        \usr/bin/robot load drive 0 slot 9`
echo "XYZ" >/tmp/zzzxxx
\usr/afsws/bin/klog utenti -pipe < /tmp/zzzxxx`
`rm /tmp/zzzxxx`
\usr/afsws/etc/backup dump server-dsks-sp08 /lunedì 0`
echo "1" > /afs/cnuce.pi.cnr.it/utenti/bin/data
set riga = \usr/bin/robot show drive 0 | awk '{print$3}'
nastro=$3
date >> /afs/cnuce.pi.cnr.it/utenti/bin/story
echo $nastro >> /afs/cnuce.pi.cnr.it/utenti/bin/story
exit
    fi
fi
fi
#####
Mercoledì
#####
    if [[ $giorno = "Wed" ]] then
        echo "XYZ" >/tmp/zzzxxx
        \usr/afsws/bin/klog utenti -pipe < /tmp/zzzxxx`
        `rm /tmp/zzzxxx`
        set riga = \usr/bin/robot show drive 0 | awk '{print$3}'
        drive=$3
        if [[ $drive = "Empty" ]] then
            \usr/bin/robot load drive 0 slot 7`
            \usr/afsws/etc/backup dump server-dsks-sp08 /lunedì/mercoledì 0 -append `
            exit
        else
            set riga = \usr/bin/robot show drive 0 | awk '{print$3}'
            nastro=$3
            $nastro in
            ANG543)
            \usr/bin/robot unload drive 0 slot 0`
            ;;
            ANG544)
            \usr/bin/robot unload drive 0 slot 1`
            ;;
            ANG545)
            \usr/bin/robot unload drive 0 slot 2`
            ;;
            ANG546)
            \usr/bin/robot unload drive 0 slot 3`
            ;;

```

```

ANG547)
`usr/bin/robot unload drive 0 slot 4`
;;
ANG548)
`usr/bin/robot unload drive 0 slot 5`
;;
ANG549)
`usr/bin/robot unload drive 0 slot 6`
;;
ANG550)
`usr/bin/robot unload drive 0 slot 7`
;;
ANG551)
`usr/bin/robot unload drive 0 slot 8`
;;
ANG552)
`usr/bin/robot unload drive 0 slot 9`
;;
esac
set riga = `usr/bin/robot show drive 0 | awk '{print$3}'`
drive=$3
if [[ $drive = "Empty" ]] then
    `usr/bin/robot load drive 0 slot 7`
    `usr/afsws/etc/backup dump server-dsks-sp08 /lunedì/mercoledì 0 -
append `
    set riga = `usr/bin/robot show drive 0 | awk '{print$3}'`
    nastro=$3
    date >> /afs/cnuce.pi.cnr.it/utenti/bin/story
    echo $nastro >> /afs/cnuce.pi.cnr.it/utenti/bin/story
        if [[ $nerr != 0 ]]
            then mail -s "Andato male" $mailaddr
        fi
    exit
fi
fi
fi
#####
# Venerdì
#####
if [[ $giorno = "Fri" ]] then
    echo "XYZ" >/tmp/zzzxxx
    `usr/afsws/bin/klog utenti -pipe < /tmp/zzzxxx`
    `rm /tmp/zzzxxx`
    set riga = `usr/bin/robot show drive 0 | awk '{print$3}'`
    drive=$3
    if [[ $drive = "Empty" ]] then
        `usr/bin/robot load drive 0 slot 8`
        usr/afsws/etc/backup dump server-dsks-sp08 /lunedì/venerdì 0 -append`
        exit
    else
        set riga = `usr/bin/robot show drive 0 | awk '{print$3}'`

```

```

nastro=$3
case $nastro in
ANG543)
`usr/bin/robot unload drive 0 slot 0`
;;
ANG544)
`usr/bin/robot unload drive 0 slot 1`
;;
ANG545)
`usr/bin/robot unload drive 0 slot 2`
;;
ANG546)
`usr/bin/robot unload drive 0 slot 3`
;;
ANG547)
`usr/bin/robot unload drive 0 slot 4`
;;
ANG548)
`usr/bin/robot unload drive 0 slot 5`
;;
ANG549)
`usr/bin/robot unload drive 0 slot 6`
;;
ANG550)
`usr/bin/robot unload drive 0 slot 7`
;;
ANG551)
`usr/bin/robot unload drive 0 slot 8`
;;
ANG552)
`usr/bin/robot unload drive 0 slot 9`
;;
esac
set riga = `usr/bin/robot show drive 0 | awk '{print$3}'`
drive=$3
if [[ $drive = "Empty" ]] then
`usr/bin/robot load drive 0 slot 8`
`usr/afsws/etc/backup dump server-dsks-sp08 /lunedì/venerdi 0 -
append`
set riga = `usr/bin/robot show drive 0 | awk '{print$3}'`
nastro=$3
date >> /afs/cnuce.pi.cnr.it/utenti/bin/story
echo $nastro >> /afs/cnuce.pi.cnr.it/utenti/bin/story
exit
fi
fi
fi

```

**Bibliografia**

---

- [1] AFS System Administrator's Guide Version 3.3, 3.4  
AFS-D4001-04  
Transarc Corporation 1993
  
- [2] AFS Command Reference Manual Version 3.3, 3.4  
AFS-D4003-05  
Transarc Corporation 1993
  
- [3] DLT™ MiniLibrary (TL891)  
User's Guide  
EK-TL891-UG. A01  
Digital Equipment Corporation 1997