Software-RAID HOWTO

Linas Vepstas, linas@linas.org v0.54 21 Novembre 1998

RAID sta per "Redundant Array of Inexpensive Disks", un modo per creare un sottosistema di dischi veloce e affidabile da una serie di dischi singoli. RAID può proteggere da un malfunzionamento del disco e può anche migliorare la situazione di un solo disco in caso di un malfunzionamento. Questo documento è un tutorial/HOWTO/FAQ per gli utenti delle estensioni MD del kernel di Linux, dei tools ad esse associati, e del loro uso. Le estensioni MD implementano RAID-0 (striping), RAID-1 (mirroring), RAID-4 e RAID-5 tramite software. Così, con le MD, non sono richiesti hardware o controller speciali per ottenere molti dei benefici della tecnologia RAID. Documentazione tradotta da Marco Meloni - eventuali suggerimenti o correzioni della traduzione italiana e contributi in italiano per l'autore possono essere inviati al traduttore: (*tonno@stud.unipg.it*)

Indice

1 Introduzione **2**

2 Considerazioni su RAID **4**

3 Considerazioni sul setup e sull'installazione **7**

4 Riparare gli errori **14**

5 Risoluzione dei problemi di installazione **19**

6 Hardware & Software Supportato **22**

7 Modificare una installazione preesistente **23**

8 Domande sulle performance, sui tool e domande stupide in genere **26**

9 RAID ad Alta Affidabilità **33**

10 Domande che attendono risposta **34**

11 Desiderata di MD e del relativo software **34**

Preambolo

RAID, anche se concepito per aumentare l'affidabilità del sistema tramite la ridondanza, può anche portare ad un falso senso di sicurezza e confidenza quando usato impropriamente. Questa falsa confidenza può portare a disastri ancora maggiori. Si noti in particolare che RAID è progettato per proteggere in caso di un funzionamento anomalo *del disco*, non in caso di un calo di tensione o di un errore dell'operatore. Cali di tensione, kernel di sviluppo contenenti errori o operatori/amministratori di sistema possono portare ad un danneggiamento non recuperabile dei dati! RAID *non* sostituisce il normale backup del sistema. Quindi sappiate cosa state facendo, fate sempre delle prove, state sul chi vive!

1 Introduzione

1. **D**: Cosa è RAID?

   R: RAID sta per Redundant Array of Inexpensive Disks, un modo per creare un sottosistema di dischi veloce e affidabile da una serie di dischi singoli. Nel mondo dei PC I è diventata iniziale di Indipendent, mentre il mercato continua a differenziare IDE e SCSI. Nel suo significato originale I significava Inexpensive (a basso costo. ndt) se comparato ad un mainframe 3380 DASD grande quanto un refrigeratore, dei drive mostruosi che facevano sembrare economiche le belle case e robetta gli anelli di diamante.

2. **D**: Cos'è questo documento?

   R: Questo documento è un tutorial/HOWTO/FAQ per gli utenti delle estensioni MD del kernel di Linux, dei tools ad esse associati e sul loro uso. Le estensioni MD implementano RAID-0 (striping), RAID-1 (mirroring), RAID-4 e RAID-5 tramite software. Con le MD, non sono così richiesti hardware o controller speciali per ottenere molti dei benefici della tecnologia RAID. Questo documento **NON È** un'introduzione alla tecnologia RAID; questa dovrete cercarla da qualche altra parte.

3. **D**: Quali livelli di RAID implementa il kernel di Linux?

   R: Striping (RAID-0) e la concatenazione lineare sono parte dei kernel della serie 2.x. La qualità di questo codice è buona (production quality); è ben compreso e ben aggiornato. È usato in diversi grandi USENET news server.

   RAID-1, RAID-4 e RAID-5 sono parte del kernel dalla versione 2.1.63 in poi. Per i kernel delle serie 2.0.x e 2.1.x , vi sono patch che forniscono questa funzione. Non sentitevi obbligati ad aggiornare il kernel alla versione 2.1.63; l'aggiornamento del kernel è difficile; è *molto* più facile applicare una patch ad un kernel precedente. La maggioranza degli utenti di RAID utilizza kernel 2.0.x, ed è su queste versioni che si è focalizzato lo sviluppo storico della tecnologia RAID. Al momento queste implementazioni sono da considerarsi in stadio di codice quasi di buona qualità; non ci sono bug noti ma ci sono dei lati poco sofisticati e dei setup di sistema non testati. Sono comunque molti gli utenti che usano il Software RAID in un ambiente di lavoro.

La funzionalità RAID-1 hot reconstruction è stata recentemente introdotta (Agosto 1997) e deve essere considerata in stadio di alfa testing. La funzionalità RAID-5 hot reconstruction sarà in alfa test prima o poi.

Una parola va spesa sulla prudenza da usare con i kernel di sviluppo della serie 2.1.x: questi sono meno stabili per molteplici ragioni. I nuovi controller di dischi (ad es. Ultra Promise) sono supportati solo nei kernel 2.1.x. Tuttavia i kernel della serie 2.1.x hanno subito molti cambiamenti del driver per i dispositivi a blocchi, nel codice DMA e interrupt, in quello per le interfacce PCI, IDE, SCSI e nel driver dei controller di dischi. La combinazione di questi fattori, abbinata ad hard disk a basso prezzo e/o a cavi di scarsa qualità può portare a discreti disastri. Il tool `ckraid`, come `fsck` e `mount` mette sotto stress il sistema RAID. Tutto questo può portare all'impossibilità di fare il boot del sistema, una situazione dove anche la combinazione magica alt-SysReq non ci salverà. Siate prudenti con i kernel 2.1.x e aspettatemi problemi. Oppure utilizzate ancora il kernel 2.0.34.

4. **D**: Utilizzo un vecchio kernel. Dove posso trovare le patch?

   R: La funzionalità RAID-0 Software e quella linear mode fanno parte di tutti i nuovi kernel di Linux. Le patch per le funzionalità Software RAID-1,4,5 sono disponibili su
   `http://luthien.nuclecu.unam.mx/~miguel/raid` . e sul quasi mirror
   `ftp://linux.kernel.org/pub/linux/daemons/raid/`
   per patch, tool e varie..

5. **D**: Dove posso trovare del materiale sulla tecnologia RAID per Linux?

   R:

   - Panoramica generale su RAID:
     `http://www.dpt.com/uraiddoc.html` .
   - Opzioni di RAID per Linux:
     `http://linas.org/linux/raid.html` .
   - Ultima versione di questo documento:
     `http://linas.org/linux/Software-RAID/Software-RAID.html` .
   - Archivio della mailing-list Linux-RAID:
     `http://www.linuxhq.com/lnxlists/` .
   - Linux Software RAID Home Page:
     `http://luthien.nuclecu.unam.mx/~miguel/raid` .
   - Linux Software RAID tools:
     `ftp://linux.kernel.org/pub/linux/daemons/raid/` .
   - Come configurare linear/stripped Software RAID:
     `http://www.ssc.com/lg/issue17/raid.html` .
   - Bootable RAID mini-HOWTO:
     `ftp://ftp.bizsystems.com/pub/raid/bootable-raid` .
   - Root RAID HOWTO:
     `ftp://ftp.bizsystems.com/pub/raid/Root-RAID-HOWTO` .
   - Linux RAID-Geschichten:
     `http://www.infodrom.north.de/~joey/Linux/raid/` .

6. **D**: Chi posso accusare di aver scritto questo documento?

   R: Questo documento è stato messo insieme da Linas Vepstas. Comunque molte informazioni e anche qualche parola sono state fornite da:

   - Bradley Ward Allen < *ulmo@Q.Net* >
   - Luca Berra < *bluca@comedia.it* >

- Brian Candler < *B.Candler@pobox.com* >
- Bohumil Chalupa < *bochal@apollo.karlov.mff.cuni.cz* >
- Rob Hagopian < *hagopiar@vu.union.edu* >
- Anton Hristozov < *anton@intransco.com* >
- Miguel de Icaza < *miguel@luthien.nuclecu.unam.mx* >
- Marco Meloni < *tonno@stud.unipg.it* >
- Ingo Molnar < *mingo@pc7537.hil.siemens.at* >
- Alvin Oga < *alvin@planet.fef.com* >
- Gadi Oxman < *gadio@netvision.net.il* >
- Vaughan Pratt < *pratt@cs.Stanford.EDU* >
- Steven A. Reisman < *sar@pressenter.com* >
- Michael Robinton < *michael@bzs.org* >
- Martin Schulze < *joey@finlandia.infodrom.north.de* >
- Geoff Thompson < *geofft@cs.waikato.ac.nz* >
- Edward Welbon < *welbon@bga.com* >
- Rod Wilkens < *rwilkens@border.net* >
- Johan Wiltink < *j.m.wiltink@pi.net* >
- Leonard N. Zubkoff < *lnz@dandelion.com* >
- Marc ZYNGIER < *zyngier@ufr-info-p7.ibp.fr* >

Copyright

Grazie a tutti di essere qui!

2 Considerazioni su RAID

1. **D**: Cosa è RAID? Perché mai dovrei usarlo?

   R: RAID è una maniera per combinare diversi disk drive in una singola unità, in modo da aumentare le prestazioni e/o l'affidabilità. Vi sono diversi tipi e implementazioni di RAID, ognuna con i suoi vantaggi e svantaggi. Per esempio, mettendo una copia degli stessi dati su due dischi (cosa chiamata **disk mirroring**, o RAID livello 1), le prestazioni in lettura possono essere migliorate leggendo alternativamente da ogni disco nel sistema di mirror. In media ogni disco è meno occupato poiché effettua solo 1/2 (nel caso di 2 dischi) o 1/3 (nel caso di tre dischi e così via) delle letture richieste. In più un sistema di mirror può aumentare l'affidabilità: se un disco si rompe, gli altri dischi contengono una copia dei dati. Differenti maniere di combinare più dischi in uno, dette **livelli RAID**, possono fornire una superiore capacità di immagazzinamento dei dati del semplice mirroring, o possono alterare i tempi di attesa (tempi di accesso), o il throughput (transfer rate), nella lettura o scrittura, offrendo comunque la ridondanza che risulta utile in caso di malfunzionamenti. **Anche se RAID può proteggere da eventuali malfunzionamenti del disco, non protegge da errori dell'operatore e dell' amministratore (errori umani), o da errori dovuti a bug di programmazione (forse dovuti anche ad errori nello stesso software RAID). La**

rete abbonda di storie tragiche di amministratori di sistema che hanno installato RAID e hanno perso tutti i loro dati. RAID non sostituisce un backup frequente e regolarmente programmato.

RAID può essere implementato via hardware, attraverso speciali controller di dischi, o via software, come un modulo del kernel che si interpone tra i driver di basso livello dei dischi e il file system. L'hardware RAID consiste comunque di un controller di dischi, una periferica alla quale si possono collegare i disk drive. Usualmente si presenta come una scheda che si può connettere su uno slot ISA/EISA/PCI/S-Bus/MicroChannel. Tuttavia qualche controller RAID è fatto per connettersi a metà del cavo che collega il normale controller ai dischi. Quelli piccoli possono entrare nell'alloggiamento di un drive; quelli grandi possono essere installati in un loro cabinet con i loro alloggiamenti per dischi e una loro alimentazione. Il più recente hardware RAID usato con le ultime e più veloci CPU fornirà probabilmente la migliore performance, ad un prezzo comunque abbastanza alto. Questo perché molti controller RAID contengono un DSP e della memoria cache che permette di togliere una fetta considerevole di carico alla CPU, e permettono un alto transfer rate grazie alla ampia memoria cache del controller. Il vecchio hardware RAID può agire come freno del sistema se usato con le nuove CPU: i vecchi DSP e memorie cache fanno da collo di bottiglia e la performance globale è spesso superata da quella del semplice RAID software e dai nuovi, normalissimi, controller. Il RAID implementato via hardware può essere vantaggioso rispetto a quello via software se può far uso della sincronizzazione dei dischi e può sapere la posizione della testina del disco rispetto al blocco del disco desiderato. È vero però che molti dischi moderni (a basso costo) non offrono questi livelli di informazione e controllo sulla loro attività e quindi molto hardware RAID non ne trae vantaggio. L'hardware RAID di diverse marche, versioni e modelli è normalmente incompatibile. Se un controller RAID si rompe, deve essere rimpiazzato da un altro controller dello stesso tipo. Al momento della stesura di questo documento (giugno 1998) un'ampia gamma di controller hardware funzionerà sotto Linux; in ogni caso nessuno di questi viene venduto con delle utilità di configurazione e gestione che funzionino sotto Linux.

Software-RAID è formato da un set di moduli del kernel, combinato con delle utilità di gestione che implementano RAID solamente tramite software e non richiedono hardware speciale. Il sottosistema RAID di Linux è implementato come strato del kernel che si pone tra i driver a basso livello dei dischi (per dischi IDE, SCSI e Paraport) e l'interfaccia del dispositivo a blocchi. Il file system, sia esso ext2fs, DOS-FAT o altro, rimane sopra l'interfaccia del dispositivo a blocchi. Software-RAID, grazie alla sua natura software, risulta essere più flessibile di una soluzione hardware. Il lato negativo è che richiede un maggiore utilizzo della CPU per poter girare bene rispetto ad una uguale implementazione hardware. Naturalmente sul costo non può essere battuto. Software-Raid ha una ulteriore importante caratteristica distintiva: opera su partizioni, perciò la partizione RAID è formata da un certo numero di partizioni di dischi. Questo contrasta con le soluzioni adottate da gran parte dell' hardware RAID, che uniscono assieme interi dischi per formarne una serie. Fatto sta che utilizzando l'hardware si ha una sottosistema RAID trasparente rispetto al sistema operativo, cosa che tende a semplificare la gestione. Con il software, ci sono molte più opzioni di configurazione e scelte, che tendono a complicare la faccenda.

Al momento della stesura di questo testo (giugno 1998) la gestione del RAID sotto Linux non è affatto banale, ed è una buona prova anche per degli esperti amministratori di sistema. La teoria delle operazioni è complessa. I tool di sistema richiedono modifiche agli script di avvio. Riprendere il controllo della situazione dopo un malfunzionamento del disco è un'operazione complessa che agevola l'errore umano. RAID non è per i novizi, ed ogni beneficio che può apportare all'affidabilità e alle prestazioni può essere facilmente controbilanciato

da una maggiore complessità. In realtà i moderni dischi sono incredibilmente affidabili e le moderne CPU e i controller sono veramente potenti. Si possono ottenere più facilmente i livelli di prestazione e l'affidabilità desiderate acquistando hardware di alta qualità e/o più veloce

2. **D**: Cosa sono i livelli RAID? Perché così tanti? Cosa li distingue?

   R: Differenti livelli RAID hanno prestazioni, ridondanza, capacità di immagazzinamento, affidabilità e costi differenti. Molti, ma non tutti, i livelli RAID offrono ridondanza per cautelarsi contro i malfunzionamenti dei dischi. Di quelli che implementano la ridondanza RAID-1 e RAID-5 sono i più popolari. RAID-1 offre migliori prestazioni mentre RAID-5 un uso più efficiente dello spazio disco a disposizione. Comunque cercare la migliore prestazione è una cosa completamente differente poiché la performance dipende fortemente da un'ampia gamma di fattori, dal tipo di applicazione alle dimensioni delle strisce, dei blocchi e dei file. Gli aspetti più difficoltosi della ricerca della migliore performance sono rimandati ad una sezione di questo HOWTO che vedremo dopo. Quanto segue descrive i differenti livelli RAID nel contesto dell'implementazione su Linux del Software RAID.

   - **RAID-linear** è una semplice concatenazione di partizioni che creano una più ampia partizione virtuale. È utile se si ha un certo numero di piccoli dischi e si vuole creare una unica, grande partizione. Questa concatenazione non offre ridondanza ed in effetti diminuisce l'affidabilità globale; la partizione così creata smetterà di funzionare non appena un solo disco si rovina.
   - **RAID-1** è chiamato anche mirroring. Due (o più) partizioni, tutte della stessa grandezza, contengono ciascuna una copia degli stessi dati che risultano disposti nella stessa maniera sulle due partizioni (in inglese disk-block by disk-block. ndt). Il mirroring offre una forte protezione contro i malfunzionamenti dei dischi: se un disco si rompe se ne ha un altro con gli stessi dati. Il mirroring può anche migliorare le prestazioni in sistemi I/O laden (con grossi carichi di input-output. ndt) poiché le richieste di lettura possono essere divise su più dischi. Sfortunatamente il mirroring è il meno efficiente in materia di capacità di immagazzinamento: due partizioni in mirroring posso immagazzinare gli stessi dati di una.
   - **Striping** è l'idea che sta dietro a tutti gli altri livelli RAID. Una striscia (in inglese stripe da cui il nome di striping. ndt) è una sequenza continua di blocchi del disco. Una striscia può essere così corta da contenere un solo blocco disco o può contenerne centinaia. I driver RAID dividono le partizioni in strisce; i vari livelli RAID differiscono nella maniera di organizzare le strisce e nei dati che vi memorizzano. Il rapporto tra le dimensioni delle strisce, la grandezza più ricorrente dei file nel file system e la loro disposizione nel disco è quello che determina le prestazioni globali di un sottosistema RAID.
   - **RAID-0** è molto simile a RAID-linear, tranne per il fatto che le partizioni che lo compongono vengono divise in strisce e quindi suddivise. Come nel caso del RAID-linear il risultato è una singola partizione virtuale. Ancora come RAID-linear, RAID-0 non offre ridondanza diminuendo quindi l'affidabilità globale: il malfunzionamento di un solo disco mette fuori uso tutto. Spesso si crede che RAID-0 abbia prestazioni migliori rispetto a RAID-linear. Questo può essere o non essere vero, dipendendo dalle caratteristiche del file system, dalla grandezza più frequente dei file comparata con la dimensione delle strisce e dal tipo di carico a cui è sottoposto. Il file system `ext2fs` dispone i file in una partizione in modo da diminuire la frammentazione. Così, per semplicità, ogni dato accesso può essere indirizzato ad uno o più dischi e quindi la suddivisione delle strisce su più dischi non offre un vantaggio apparente. Comunque sia vi sono delle differenze nelle prestazioni, dipendenti dai dati, dal carico di lavoro e dalla dimensione delle strisce.

- **RAID-4** suddivide in strisce come RAID-0, ma richiede una partizione aggiuntiva per memorizzare le informazioni sulla parità. La parità è usata per offrire ridondanza sui dati: se un disco si rovina i dati sui dischi rimanenti possono essere usati per ricostruire quelli che erano sul disco rotto. Dati N dischi di dati e un disco di parità, la striscia di parità è computata prendendo una striscia da ognuno dei dischi di dati ed effettuando un XOR tra di esse. Quindi la capacità di memorizzazione di una serie di (N+1) dischi RAID-4 è N, molto meglio del mirroring di (N+1) dischi e buona come un setup RAID-0, per N grande. Da notare che per N=1 vi è un disco di dati e un disco di parità e RAID-4 somiglia molto al mirroring, nel quale ognuno dei due dischi è la copia dell'altro. RAID-4 **NON** offre le prestazioni in lettura/scrittura del mirroring ed anzi la sua performance in scrittura è considerevolmente peggiore. In breve questo accade a causa del fatto che l'aggiornamento della parità richiede la lettura della vecchia parità prima che la nuova parità venga calcolata e scritta. In ambienti con un grosso carico di scrittura il disco di parità può diventare un collo di bottiglia poiché ogni processo di scrittura deve accedere al disco di parità.
- **RAID-5** evita il collo di bottiglia in scrittura di RAID-4 memorizzando la striscia di parità su ognuno dei dischi. Ovviamente la prestazione in scrittura non è ancora buona come quella del mirroring, visto che la striscia di parità deve anche qui essere letta e deve esservi effettuato lo XOR prima che sia scritta. Anche la prestazione in lettura non è buona come quella del mirroring poiché, dopo tutto, vi è una sola copia dei dati e non due o più. Il vantaggio principale di RAID-5 sul mirroring è che offre ridondanza e protezione nel caso di malfunzionamento di un solo disco, e allo stesso tempo ha una capacità di memorizzazione molto più alta quando è usato con tre o più drive.
- **RAID-2 e RAID-3** sono usati raramente, e sono stati in qualche maniera resi obsoleti dalla moderna tecnologia dei dischi. RAID-2 è simile a RAID-4, ma memorizza informazioni ECC al posto della parità. Poiché tutti i dischi moderni incorporano sotto sotto un controllo ECC, il vantaggio è minimo. RAID-2 può dare una maggiore coerenza ai dati se viene a mancare la corrente mentre è in corso un'operazione di scrittura; però un gruppo di continuità e uno shutdown pulito danno gli stessi vantaggi. RAID-3 è simile a RAID-4 tranne per il fatto che usa la minore grandezza possibile per le strisce. Il risultato è che ogni operazione di lettura interessa tutti i dischi, facendo diventare difficile/impossibile soddisfare richieste di I/O contemporanee. Per evitare il ritardo dovuto alla latenza rotazionale RAID-3 richiede che la rotazione di tutti i dischi possa essere sincronizzata. Molto hardware moderno non dispone della capacità di sincronizzazione o, se ne dispone, mancano i connettori necessari, i cavi e la documentazione di chi lo ha prodotto. Né RAID-2 né RAID-3 sono livelli RAID supportati dai driver di Software RAID per Linux.
- **Altri livelli RAID** sono stati definiti da vari ricercatori e produttori. Molti di questi non sono altro che la sovrapposizione di un tipo di raid su un altro. Qualcuno richiede dell'hardware speciale e altri sono protetti da brevetto. Non vi è una nomenclatura universalmente accettata per questi altri livelli. A volte i vantaggi di questi altri sistemi sono piccoli o almeno non appaiono finché il sistema non è sottoposto ad un alto livello di stress. Eccettuata la sovrapposizione di RAID-1 su RAID-0/linear, il Software RAID per Linux non consente nessuna di queste altre variazioni.

3 Considerazioni sul setup e sull'installazione

1. **D**: Quale è il modo migliore di configurare Software RAID?

R: Continuamente riscopro il fatto che la pianificazione del file-system è uno dei lavori più difficili sotto Unix. Per rispondere alla domanda, posso descrivere cosa si può fare.

Supponiamo il setup che segue:

- 2 dischi EIDE, da 2.1 Gb ciascuno.

disco	partizione	montata su	dimensione	dispositivo
1	1	/	300M	/dev/hda1
1	2	swap	64M	/dev/hda2
1	3	/home	800M	/dev/hda3
1	4	/var	900M	/dev/hda4
2	1	/root	300M	/dev/hdc1
2	2	swap	64M	/dev/hdc2
2	3	/home	800M	/dev/hdc3
2	4	/var	900M	/dev/hdc4

- Ogni disco è su un controller (e cavo) separato. La mia teoria è che un guasto al controller o al cavo non manderà in tilt tutti e due i dischi. Questo permette un miglioramento delle performance rispetto alla gestione di operazioni parallele
- Installare Linux su (/) della partizione `/dev/hda1`. Marcare questa partizione come avviabile.
- `/dev/hdc1` conterrà una copia "fredda" di `/dev/hda1`. Questa non è una copia RAID, è proprio una semplice copia. Serve solamente nel caso che il primo disco si rompa; si può usare un disco, di recupero, marcare `/dev/hdc1` come avviabile, e usare questa partizione per continuare a lavorare senza dover reinstallare il sistema. Può anche essere utile mettere una copia del kernel di `/dev/hdc1` su LILO per semplificare il boot in caso di malfunzionamento.

  Qui si suppone che nel caso di un grave malfunzionamento si possa ancora far partire il sistema senza preoccupazioni riguardanti la corruzione dei superblock raid o di altri errori del raid che non si capiscono.
- `/dev/hda3` e `/dev/hdc3` saranno il mirror `/dev/md0`.
- `/dev/hda4` e `/dev/hdc4` saranno il mirror `/dev/md1`.
- abbiamo scelto`/var` e `/home` per essere mirrorate in partizioni separate, seguendo questa logica
  - / (la partizione root) conterrà dati relativamente statici, non soggetti a cambiamenti. Ad ogni effetto sarà si sola lettura anche se non sarà impostata e montata veramente in sola lettura
  - `/home` conterrà i dati che cambiano lentamente.
  - `/var` conterrà i dati che cambiano rapidamente, inclusi i mail spool, i database ed i log del web server.

  L'idea che sta dietro all'uso di più partizioni separate è che **se**, per qualche bizzarra, ragione che sia un errore umano, un calo di tensione, o altro il sistema operativo impazzisce, il danno è limitato ad una sola partizione. In un caso tipico vi è un calo di tensione mentre il sistema sta scrivendo sul disco. Questo lascerà sicuramente il file system in uno stato di inutilizzabilità, a cui sarà posto rimedio da `fsck` nel boot seguente. Anche se `fsck` farà del suo meglio per rimettere a posto la situazione evitando di fare ulteriori danni, è confortante sapere che ogni danno è stato limitato ad una sola partizione. In un altro caso tipico l'amministratore di sistema fa un errore durante le operazioni di recupero del file system, cosa che porta alla cancellazione o alla distruzione dei dati. L'uso delle partizioni può aiutare a contenere le ripercussioni degli errori dell'operatore.

- Un'altra scelta ragionevole per la disposizione delle partizioni potrebbe essere quella di `/usr` o di `/opt`. In effetti, `/opt` e `/home` sono una buona scelta come partizioni RAID-5, se si hanno più hard disk. Una parola sulla prudenza da utilizzare: **NON** mettere `/usr` in una partizione RAID-5. Se si avesse un grave malfunzionamento si potrebbe scoprire che non si può montare `/usr`, e che si ha bisogno di qualche strumento che vi risiede (ad es. i programmi per la rete, o il compilatore.) Con RAID-1, se si ha un malfunzionamento e RAID smette di funzionare, si può almeno montare uno dei due mirror. La stessa cosa non si può fare con ogni altro livello RAID (RAID-5, striping, linear RAID).

Così, per rispondere alla domanda:

- installare il S.O. sul disco 1, partizione 1. NON montare altre partizioni.
- installare RAID seguendo le istruzioni.
- configurare `md0` e `md1`.
- convincersi del fatto che si sa che cosa fare in caso di malfunzionamento del disco! Si scoprano gli errori dell'amministratore di sistema adesso, non non durante una vera crisi! Sperimentate! (noi abbiamo tolto corrente durante l'attività del disco, poco ortodosso ma fa imparare molto).
- effettuate una successione di mount/copy/unmount/rename/reboot per muovere `/var` su `/dev/md1`. Fatelo attentamente, non è pericoloso.
- godetevi il tutto!

2. **D**: Quale è la differenza tra i comandi `mdadd`, `mdrun`, *etc.* e quelli `raidadd`, `raidrun`?

   R: I nomi dei tool sono stati cambiati dalla versione 0.5 del pacchetto raidtools. La convenzione che voleva che i comandi iniziassero per `md` era usata nella versione 0.43 e precedenti, mentre quella nuova che fa iniziare i comandi per `raid` viene usata nella versione 0.5 e successive.

3. **D**: Vorrei utilizzare RAID-Linear/RAID-0 presente nel kernel 2.0.34. Vorrei non applicare la patch raid, poiché non sono richieste per RAID-0/Linear. Dove posso procurarmi i tool RAID per gestire il sistema?

   R: Questa è una bella domanda, poiché i più nuovi tool raid abbisognano che le patch RAID-1,4,5 siano state applicate al kernel per compilarsi. Non conosco versioni binarie, precompilate dei tool raid disponibili in questo momento. Comunque sia, esperimenti hanno dimostrato che i file binari dei tool raid, compilati su un kernel 2.1.100, sembrano funzionare abbastanza bene nella creazione di una partizione RAID-0/linear sotto 2.0.34. Un'anima impavida me li ha chiesti ed io ho **temporaneamente** reso disponibili i binari di mdadd, mdcreate, ecc, su http://linas.org/linux/Software-RAID/ Avrete bisogno delle pagine di manuale, ecc. che si trovano nel pacchetto standard dei raid-tools.

4. **D**: Posso mettere in strip/mirror la partizione di root (`/`)? Perché non posso far partire Linux direttamente dai dischi `md`?

   R: Sia LILO che Loadlin hanno bisogno di una partizione che non sia in strip/mirror per potervi leggere l'immagine del kernel. Se volete mettere in strip/mirror la partizione di root (`/`), avrete bisogno di creare una partizione che non sia in strip/mirror dove poter mettere il kernel. Tipicamente questa partizione viene chiamata `/boot`. Fatto questo si può quindi usare o il supporto per il ramdisk iniziale (initrd) o le patch di Harald Hoyer < *HarryH@Royal.Net* > che consentono ad una partizione in strip/mirror di essere usata come root. (Adesso queste patch sono una parte standard dei recenti kernel 2.1.x)

Si possono usare approcci diversi. Uno è documentato dettagliatamente nel Bootable RAID mini-HOWTO:

`ftp://ftp.bizsystems.com/pub/raid/bootable-raid` .

In alternativa, si può usare `mkinitrd` per costruire un'immagine per il ramdisk, vedere di seguito.

Edward Welbon < *welbon@bga.com* > ha scritto:

- ... tutto quello che serve è uno script che gestisca il setup di boot. Per montare un file system `md` come root, la cosa principale da fare è costruire un'immagine iniziale del file system che abbia i moduli e i tool md necessari per far partire `md`. Io ho un semplice script che fa tutto ciò.
- Come supporto per il boot utilizzo un piccolo ed **economico** disco SCSI (170MB lo ho preso usato per $20). Il disco è collegato ad un AHA1452, ma avrebbe potuto essere un disco IDE a buon prezzo sull'interfaccia nativa IDE. Non c'è bisogno che sia un disco molto veloce poiché serve solamente per il boot.
- Questo disco contiene un piccolo file system dove trova posto il kernel e l'immagine del file system per `initrd`. L'immagine iniziale del file system contiene abbastanza roba da permettermi di caricare il modulo driver per il dispositivo raid SCSI e poter accedere alla partizione che diverrà root. Poi eseguo un

  ```
  echo 0x900 > /proc/sys/kernel/real-root-dev
  ```

  (`0x900` è per `/dev/md0`) e esco da `linuxrc`. Il boot procede normalmente da qui in poi.
- Ho compilato molti driver come moduli eccetto quello per l'AHA1452 che serve per il file system `initrd`. Il kernel che uso è così molto piccolo. Il metodo è perfettamente affidabile, lo sto usando sin da prima della versione 2.1.26 e non ho mai avuto un problema che non sia riuscito a risolvere facilmente. Il file system è addirittura sopravvissuto a diversi crash dei kernel 2.1.4[45] senza reali difficoltà.
- Una volta avevo partizionato i dischi raid in maniera tale che i cilindri iniziali del primo disco raid contenevano il kernel e i cilindri iniziali del secondo disco raid contenevano l'immagine iniziale del file system, adesso invece ho messo la swap nei primi cilindri dei dischi raid poiché questi sono i più veloci (perché sprecarli nel boot?).
- La cosa bella dell'avere un dispositivo che costa poco dedicato al boot è che è facile effettuarne il boot e che risulta utile come disco di recupero se necessario. Se siete interessati, potete dare un'occhiata allo script che genera l'immagine iniziale per il ram disk e che quindi fa partire `LILO`.

  `http://www.realtime.net/~welbon/initrd.md.tar.gz`

  Per adesso è abbastanza per fare quello che serve. Non è specificamente bello e potrebbe sicuramente costruire un'immagine del file system molto più piccola per il ram disk iniziale. Dovrebbe essere facile renderlo più efficiente. Ma del resto usa `LILO` così com'è. Se gli apportate dei miglioramenti vi prego di mandarmene una copia. 8-)

5. **D**: Ho sentito dire che si può usare il mirroring sullo striping RAID. È vero? Posso usare il mirroring sul dispositivo di loopback?

R: Si, ma non il contrario. Si può mettere una striscia su più dischi e poi effettuarne il mirroring. Comunque sia lo striping non può essere messo al di sopra del mirroring.

Per darne una breve spiegazione tecnica si può dire che le personality linear e stripe usano la routine `ll_rw_blk` per gli accessi. La routine `ll_rw_blk` mappa dispositivi di dischi e settori, non blocchi. I dispositivi a blocchi possono essere stratificati uno sull'altro; ma i dispositivi che effettuano un accesso al disco diretto, a basso livello, come fa `ll_rw_blk` non possono essere sovrapposti.

In questo momento (Novembre 1997) RAID non può funzionare sui dispositivi di loopback, anche se questo dovrebbe essere possibile a breve.

6. **D**: Ho due piccoli dischi e tre dischi più grandi. Posso concatenare i due dischi piccoli con RAID-0 e quindi creare una partizione RAID-5 con questi e quelli più grandi?

   R: In questo momento (Novembre 1997), non si può creare una partizione RAID-5 in questa maniera. Si può farlo solo con RAID-1 al di sopra della concatenazione dei drive.

7. **D**: Quale è la differenza tra RAID-1 e RAID-5 per una configurazione che prevede due dischi (cioè la differenza tra una serie di due dischi RAID-1 e una serie di due dischi RAID-5)?

   R: Non c'è differenza nella capacità di immagazzinamento. Non si possono aggiungere dischi a nessuno dei due sottosistemi per aumentarne la capacità (vedi la domanda qui di seguito per i dettagli).

   RAID-1 offre un vantaggio nella prestazione in lettura: il driver RAID-1 usa la tecnologia distributed-read (lettura distribuita. ndt) per leggere contemporaneamente due settori, uno da ogni disco, raddoppiando la performance in lettura.

   Il driver RAID-5, anche se fortemente ottimizzato, attualmente (Settembre 1997) non considera il fatto che il disco di parità è una copia del disco dati. Quindi le letture avvengono in maniera seriale.

8. **D**: Come posso proteggermi dal malfunzionamento di due dischi?

   A: Qualcuno degli algoritmi RAID protegge da un malfunzionamento multiplo dei dischi, ma nessuno di questi algoritmi è attualmente implementato da Linux. Detto ciò il Software RAID per Linux può essere utilizzato per proteggersi da un malfunzionamento di più dischi stratificando serie su serie di dischi. Per esempio, nove dischi possono essere utilizzati per creare tre serie raid-5. Quindi queste tre serie possono a loro volta essere legate assieme in una singola serie di dischi RAID-5. In effetti questo tipo di configurazione arriva a proteggere da un malfunzionamento di tre dischi. Va notato il fatto che una grande quantità di spazio disco va "persa" per la ridondanza delle informazioni.

   ```
   Per una serie di NxN dischi raid-5
   N=3, 5 dischi su 9 sono usati per la parità (=55%)
   N=4, 7 dischi su 16
   N=5, 9 dischi su 25
   ...
   N=9, 17 dischi su 81 (=~20%)
   ```

   In generale, una serie di MxN dischi userà MxN-1 dischi per la parità. Lo spazio perso è minimo quando M=N. Un'altra alternativa è quella di creare una serie RAID-1 con tre dischi. Va notato il fatto che poiché tutti e tre i dischi contengono dati identici, 2/3 dello spazio vanno "sprecati".

9. **D**: Mi piacerebbe capire come è possibile che ci sia un programma tipo `fsck`: se la partizione non è stata smontata in maniera ortodossa, `fsck` interviene e riaggiusta il filesystem da solo in più del 90% dei casi. Poiché la macchina è capace di porsi rimedio da sola con `ckraid -fix`, perché non farlo diventare automatico?

   R: Si può ottenere ciò aggiungendo linee come le seguenti a `/etc/rc.d/rc.sysinit`:

   ```
   mdadd /dev/md0 /dev/hda1 /dev/hdc1 || {
       ckraid --fix /etc/raid.usr.conf
       mdadd /dev/md0 /dev/hda1 /dev/hdc1
   ```

```
}
```

o

```
mdrun -p1 /dev/md0
if [ $? -gt 0 ] ; then
        ckraid --fix /etc/raid1.conf
        mdrun -p1 /dev/md0
fi
```

Prima di presentare uno script più completo e affidabile, rivediamo la teoria delle operazioni.

Gadi Oxman ha scritto: In uno shutdown sporco, Linux può rimanere in uno degli stati seguenti:

- Il dischi RAID erano stati aggiornati con i dati contenuti nella memoria cache a loro destinata quando è avvenuto lo shutdown; non sono andati persi dati.
- La memoria cache dei dischi RAID conteneva informazioni non scritte sui dischi quando è avvenuto il blocco del sistema; questo ha portato ad un filesystem danneggiato e potenzialmente alla perdita di dati.

  Quest'ultimo stato può essere ulteriormente suddiviso in altri due stati:

  - Linux era in fase di scrittura dati quando si è avuto lo shutdown.
  - Linux non era in fase di scrittura dati quando si è verificato il blocco.

Supponiamo che stavamo usando una serie di dischi RAID-1. Nel caso (2a) potrebbe accadere che, prima del blocco, un piccolo numero di blocchi dati sia stato scritto con successo su solo alcuni dei dischi di mirror e al prossimo boot i mirror non conterranno più' gli stessi dati.

Se si ignorassero le differenze dei mirror, il codice di bilanciamento della lettura dei raidtools-0.36.3 potrebbe scegliere di leggere i suddetti dati da uno qualsiasi dei dischi di mirror, cosa che porterebbe ad un comportamento incoerente (per esempio, l'output di `e2fsck -n /dev/md0` potrebbe essere differente di volta in volta).

Poiché RAID non protegge dagli shutdown sporchi, usualmente non c'e' un modo "sicuramente corretto" di correggere le differenze nei dischi di mirror e il danneggiamento del filesystem.

Per esempio il comportamento predefinito di `ckraid -fix` sarà quello di scegliere il primo disco di mirror operativo e aggiornare gli altri dischi di mirror con il suo contenuto. Tuttavia, a seconda della situazione dei dischi al momento del blocco, i dati negli altri dischi di mirror potrebbero essere più recenti e si potrebbe scegliere di copiare i dati da quei dischi o forse di usare un metodo differente per riparare le cose.

Lo script che segue definisce una delle più robuste sequenze di boot. In particolare si cautela dalle lunghe ripetizioni dell'esecuzione di `ckraid` quando si ha a che fare con dischi, controller o driver dei controller che non cooperano. Lo si modifichi in modo da adeguarlo alla propria configurazione, e lo si copi su `rc.raid.init`. Quindi si esegua `rc.raid.init` dopo che la partizione di root è stata controllata da fsck e montata in lettura/scrittura ma prima che le rimanenti partizioni siano controllate da fsck. Assicurarsi che la directory attuale sia nel percorso di ricerca.

```
mdadd /dev/md0 /dev/hda1 /dev/hdc1 || {
    rm -f /fastboot             # forza l'esecuzione di fsck
    ckraid --fix /etc/raid.usr.conf
    mdadd /dev/md0 /dev/hda1 /dev/hdc1
}
```

```
# se il sistema si bloccasse più avanti durante questo processo di boot
# vorremmo che almeno questo dispositivo md non ne risentisse.
 /sbin/mdstop /dev/md0

mdadd /dev/md1 /dev/hda2 /dev/hdc2 || {
    rm -f /fastboot             # forza l`esecuzione di fsck
    ckraid --fix /etc/raid.home.conf
    mdadd /dev/md1 /dev/hda2 /dev/hdc2
}
# se il sistema si bloccasse più avanti durante questo processo di boot
# vorremmo che almeno questo dispositivo md non ne risentisse.
/sbin/mdstop /dev/md1

mdadd /dev/md0 /dev/hda1 /dev/hdc1
mdrun -p1 /dev/md0
if [ $? -gt 0 ] ; then
    rm -f /fastboot             #  forza l`esecuzione di fsck
    ckraid --fix /etc/raid.usr.conf
    mdrun -p1 /dev/md0
fi
# se il sistema si bloccasse più avanti durante questo processo di boot
# vorremmo che almeno questo dispositivo md non ne risentisse.
/sbin/mdstop /dev/md0

mdadd /dev/md1 /dev/hda2 /dev/hdc2
mdrun -p1 /dev/md1
if [ $? -gt 0 ] ; then
    rm -f /fastboot             #   forza l`esecuzione di fsck
    ckraid --fix /etc/raid.home.conf
    mdrun -p1 /dev/md1
fi
# se il sistema si bloccasse più avanti durante questo processo di boot
# vorremmo che almeno questo dispositivo md non ne risentisse.
/sbin/mdstop /dev/md1

# OK, adesso con i soli comandi md. Se ci fossero stati errori
# i controlli precedenti dovrebbero aver rimesso tutto a posto.
/sbin/mdadd /dev/md0 /dev/hda1 /dev/hdc1
/sbin/mdrun -p1 /dev/md0

/sbin/mdadd /dev/md12 /dev/hda2 /dev/hdc2
/sbin/mdrun -p1 /dev/md1
```

In aggiunta a questo si dovrà creare un file `rc.raid.halt` che dovrebbe apparire come questo:

```
/sbin/mdstop /dev/md0
/sbin/mdstop /dev/md1
```

Assicuratevi di aver modificato sia `rc.sysinit` che `init.d/halt` per far eseguire questa procedura da qualsiasi parte il filesystem venga smontato prima di un halt/reboot. (Si noti che `rc.sysinit` smonta ed effettua un reboot se `fsck` termina l'esecuzione con un errore.)

10. **D**: Posso configurare metà di un mirror RAID-1 con il solo disco che ho adesso e poi dopo aggiungervi semplicemente un altro disco?

    R: Con gli strumenti di adesso no, almeno non in maniera semplice. In particolare non si può solamente copiare il contenuto di un disco su un altro e poi appaiarli. Questo a causa del fatto che i driver RAID usano un poco di spazio alla fine della partizione per memorizzare i superblock. Questo diminuisce leggermente lo spazio disponibile per il filesystem; ma se si provasse a forzare una partizione RAID-1 su una partizione con un filesystem preesistente, i superblock sovrascriverebbero una parte del filesystem confondendo i dati. Poiché il filesystem ext2fs distribuisce i file in maniera casuale su una partizione (per evitarne la frammentazione), con grossa probabilità qualche file risiederà alla fine della partizione anche se il disco non è pieno.

    Se siete abili, suppongo che possiate calcolarvi quanto spazio il superblock RAID occuperà e quindi rendere il filesystem leggermente più piccolo, in modo da lasciare lo spazio di cui la memorizzazione del superblock RAID avrà bisogno in seguito. Ma, se siete così abili, sarete quindi abbastanza bravi da modificare i tool in modo tale che lo facciano automaticamente (i tool non sono terribilmente complessi).

    Nota: il lettore attento avrà pensato che il trucco seguente potrebbe funzionare; non l'ho provato né verificato: Eseguite `mkraid` con `/dev/null` come uno dei dispositivi. Quindi eseguite `mdadd -r` sul solo vero disco (non eseguite mdadd `/dev/null`). Il comando `mkraid` dovrebbe aver configurato con successo il sistema raid, e il comando mdadd serve solo a forzare il funzionamento del sistema in modalità degradata (in inglese degraded mode. ndt), come se uno dei due dischi fosse rotto.

4 Riparare gli errori

1. **D**: Lavoro con un dispositivo RAID-1 (mirroring) e la corrente è andata via mentre il disco era in attività. Cosa devo fare?

    R: La ridondanza che i livelli RAID offrono serve a proteggere nei confronti di un malfunzionamento del **disco**, non contro un difetto di **alimentazione**.

    Vi sono diversi modi di rimettere le cose a posto in questa situazione.

    - Metodo (1): Usare i tool raid. Questi possono essere usati per rimettere in sincronia il sistema raid. I danni al filesystem non vengono corretti; dopo che il sistema raid è stato rimesso in sincronia, il filesystem deve ancora essere riparato con fsck. I sistemi RAID possono essere controllati con `ckraid /etc/raid1.conf` (per RAID-1, altrimenti `/etc/raid5.conf`, etc.)

      Eseguendo `ckraid /etc/raid1.conf -fix` il programma sceglierà uno dei dischi della serie (usualmente il primo), e userà questo come copia master, copiando i suoi blocchi su quelli degli altri dischi nel mirror. Per designare un disco da utilizzare come copia master si può usare l'opzione `-force-source`: per esempio, `ckraid /etc/raid1.conf -fix -force-source /dev/hdc3` . Il comando ckraid può essere lanciato senza l'opzione `-fix` per verificare il sistema RAID inattivo senza apportargli modifiche. Quando vi sentirete a vostro agio con le modifiche proposte, potrete aggiungere l'opzione `-fix`.
    - Metodo (2): Paranoico, spenditempo, non molto migliore del primo metodo. Assumiamo che la serie di dischi RAID sia formata da due dischi, e consista delle partizioni `/dev/hda3` e `/dev/hdc3`. Potete provare ciò che segue:

(a) `fsck /dev/hda3`

(b) `fsck /dev/hdc3`

(c) si decida quale delle due partizioni ha meno errori o dove si possano aggiustare meglio o dove sono i dati che vi interessano. Scegliete l'una o l'altra come nuova copia "master". Diciamo che avete scelto `/dev/hdc3`.

(d) `dd if=/dev/hdc3 of=/dev/hda3`

(e) `mkraid raid1.conf -f -only-superblock`

Al posto degli ultimi due passi, potete eseguire `ckraid /etc/raid1.conf -fix -force-source /dev/hdc3` che dovrebbe essere leggermente più veloce.

- Metodo (3): Versione del metodo precedente per pigri. Se non avete voglia di aspettare che il lungo controllo di fsck venga completato, va bene anche saltare i primi tre passi illustrati sopra e andare direttamente agli ultimi due. Assicuratevi solamente che venga eseguito `fsck /dev/md0` dopo che avete fatto. Il metodo (3) è solo il metodo (1) travestito.

In ogni caso i passi precedenti serviranno solo a sincronizzare i sistemi raid. Il filesystem probabilmente avrà ancora bisogno di riparazioni: perciò fsck dovrà essere eseguito sul dispositivo md quando esso è attivo e non è stato montato.

Con una serie di tre dischi RAID-1, vi sono più possibilità come quella di usare due dischi per "votare" una risposta a maggioranza. I tool per automatizzare questa procedura attualmente (Settembre 97) non esistono.

2. **D**: Ho un sistema RAID-4 o RAID-5 (parità) e la corrente è andata via mentre i dischi erano in attività. Cosa devo fare adesso?

R: La ridondanza che i livelli RAID offrono serve a proteggere nei confronti di un malfunzionamento del **disco**, non contro un difetto di **alimentazione**.

Poiché i dischi formanti una serie RAID-4 o RAID-5 non contengono un filesystem che fsck può leggere vi sono meno scelte nella riparazione. Non si può usare fsck per un controllo o una riparazione preliminare; si deve usare prima ckraid.

Il comando `ckraid` può essere eseguito in modalità sicura senza l'opzione `-fix` per verificare il sistema RAID senza apportargli cambiamenti. Quando vi sentirete a vostro agio con i cambiamenti proposti basterà aggiungere l'opzione `-fix`.

Se si vuole si può provare designando uno dei dei dischi come "disco rotto". Fatelo con l'opzione `-suggest-failed-disk-mask` (in inglese suona come suggerisci-maschera-disco-rotto. ndt).

Un solo bit dovrà essere indicato nell'opzione: RAID-5 non può recuperare due dischi non funzionanti. La maschera è una maschera binaria: quindi:

```
0x1 == primo disco
0x2 == secondo disco
0x4 == terzo disco
0x8 == quarto disco, etc.
```

In alternativa si può scegliere di modificare i settori di parità usando l'opzione `-suggest-fix-parity`. Questa farà si che la parità venga ricalcolata dagli altri settori.

Le opzioni `-suggest-failed-disk-mask` e `-suggest-fix-parity` possono essere usate senza problemi per la sola verifica. Non vengono apportati cambiamenti se l'opzione `-fix` non è stata specificata. Quindi potete sperimentare schemi differenti di riparazione.

3. **D**: Il mio dispositivo RAID-1, `/dev/md0` è formato da due partizioni di hard disk: `/dev/hda3` e `/dev/hdc3`. Recentemente, il disco che conteneva `/dev/hdc3` si è rotto, ed è stato rimpiazzato da

un nuovo disco. Il mio migliore amico, che non conosce RAID, dice che la cosa corretta da fare adesso è "`dd if=/dev/hda3 of=/dev/hdc3`". Ho provato a farlo, ma le cose continuano a non funzionare.

R: Il tuo migliore amico dovrebbe rimanere alla larga dal tuo computer. Fortunatamente non vi sono stati danni gravi. Si può riaggiustare il tutto eseguendo:

```
mkraid raid1.conf -f --only-superblock
```

Usando `dd`, sono state create due copie identiche della partizione. Questo va quasi bene, tranne per il fatto che le estensioni RAID-1 del kernel si aspettano che i superblock RAID siano differenti. Così, quando si prova a riattivare RAID, il software nota il problema e disattiva una delle due partizioni. Ricreando i superblock, si dovrebbe avere un sistema perfettamente funzionante.

4. **D**: La mia versione di `mkraid` non ha un'opzione `-only-superblock` . Che devo fare?

   R: I nuovi tool non supportano questa opzione, che è stata rimpiazzata da `-force-resync`. È stato riferito che la seguente sequenza di comandi funziona con gli ultimi tool e software:

   ```
   umount /web (dove /dev/md0 è stata montata)
   raidstop /dev/md0
   mkraid /dev/md0 --force-resync --really-force
   raidstart /dev/md0
   ```

   Dopo questo, un `cat /proc/mdstat` dovrebbe dare `resync in progress`, e a questo punto dovrebbe essere possibile effettuare un `mount /dev/md0`.

5. **D**: Il mio dispositivo RAID-1, `/dev/md0` è formato da due partizioni: `/dev/hda3` e `/dev/hdc3`. Il mio migliore amico/a (in inglese My best (girl)friend NdT), che non conosce RAID, ha eseguito `fsck` su `/dev/hda3` mentre io non guardavo e adesso RAID ha smesso di funzionare. Cosa devo fare?

   R: Il concetto di migliore amico andrebbe riesaminato. In generale, `fsck` non dovrebbe essere mai eseguito su una singola partizione facente parte di un sistema RAID. Presumendo che nessuna partizione sia stata fortemente danneggiata non sono andati persi dati e il dispositivo RAID-1 può essere recuperato come segue:

   (a) effettuare un backup del filesystem di `/dev/hda3`

   (b) `dd if=/dev/hda3 of=/dev/hdc3`

   (c) `mkraid raid1.conf -f -only-superblock`

   Questo dovrebbe riportare al funzionamento il dispositivo di mirror.

6. **D**: Perché la procedura precedente funziona?

   R: Perché ogni partizione componente un mirror RAID-1 è una copia perfettamente valida del filesystem. Addirittura il mirroring può essere disabilitato e una delle partizioni può venire montata e funzionerà senza problemi come un filesystem normale, senza RAID. Quando si è pronti a ripartire con RAID-1, si smonti la partizione e si seguano le istruzioni sopracitate per far ripartire il mirror. Si noti che le istruzioni di cui sopra valgono SOLO per RAID-1 e per nessun altro livello.

   Vi potrebbe far sentire più a vostro agio l'invertire la direzione della copia di cui sopra: copiare **dal** disco che non è stato toccato **a** quello che lo è stato. Solo assicuratevi che alla fine venga eseguito fsck sul risultante dispositivo md.

7. **D**: Mi sento un po' confuso dalle domande riportate sopra, ma ancora non mollo. Ma è sicuro eseguire `fsck /dev/md0` ?

R: Si, è una cosa sicura l'esecuzione di `fsck` sui dispositivi `md`. In effetti questo è `l'unico` posto sicuro dove eseguire `fsck`.

8. **D**: Se un disco si sta deteriorando lentamente, sarà ovvio scoprire quale sia? Sono preoccupato dal fatto che non lo sia, e questa confusione potrebbe portare a qualche decisione pericolosa da parte di un amministratore di sistema.

   R: Una volta che un disco si rompe, un codice di errore viene trasmesso dal driver a basso livello al driver RAID. Il driver RAID marcherà questo disco come "cattivo" nei superblock dei dischi "buoni" (in questa maniera più avanti sapremo quale dei dischi di mirror sia buono e quale non) e continuerà le operazioni RAID sui rimasti dischi funzionanti.

   Questo, va da sé, dà per certo che il disco ed i driver di basso livello possano accorgersi di un errore in lettura/scrittura e che, per esempio, non continuino a danneggiare i dati in silenzio. Questo è vero per i driver attuali (schemi di rilevazione dell'errore sono usati internamente) ed è base delle operazioni RAID.

9. **D**: E sulla riparazione a caldo?

   R: Si sta lavorando per completare la "ricostruzione a caldo". Con questa funzionalità, si possono aggiungere dei dischi "spare" al set RAID (sia esso di livello 1 o 4/5), e quando un disco smette di funzionare i dati contenuti in esso saranno ricostruiti su uno dei dischi spare durante l'attività, senza nemmeno aver bisogno di fermare il set RAID.

   Tuttavia, per usare questa funzionalità, il disco spare deve essere stato dichiarato durante il boot, o esso dovrà essere aggiunto "a caldo", cosa che richiede l'uso di connettori e cabinet speciali che consentano di aggiungere un disco senza togliere corrente.

   Da Ottobre 97 esiste una versione beta di MD che consente:

   - la ricostruzione dei livelli RAID 1 e 5 su dischi spare
   - la ricostruzione della parità di RAID-5 dopo uno shutdown sporco
   - l'aggiunta "a caldo" di dischi spare ad una serie già in funzione di dischi RAID 1 o 4/5.

   Come predefinizione, la ricostruzione automatica è in questo momento (Dicembre 97) disabilitata a causa della natura ancora sperimentale di questo lavoro. Può comunque essere abilitata cambiando il valore di `SUPPORT_RECONSTRUCTION` in `include/linux/md.h`.

   Se i drive spare sono stati configurati su un sistema raid quando esso è stato creato e la ricostruzione è stata attivata nel kernel, i drive spare conterranno da subito il superblock RAID (scritto da `mkraid`), e il kernel sarà capace di ricostruire il contenuto in maniera automatica (senza bisogno degli usuali passi `mdstop`, cambia il drive,`ckraid`, `mdrun`).

   Se non avete configurato la ricostruzione automatica e non avete configurato un disco spare, la procedura descritta da Gadi Oxman < *gadio@netvision.net.il* > è quella raccomandata:

   - Per adesso, una volta che il primo disco è stato rimosso, il sistema RAID continuerà a funzionare in modalità "degradata". Per riportarlo alla piena funzionalità avrete bisogno di:
     - fermare il sottosistema RAID (`mdstop /dev/md0`)
     - rimpiazzare il disco rotto
     - eseguire `ckraid raid.conf` per ricostruire il suo contenuto
     - far ripartire RAID (`mdadd`, `mdrun`).

     A questo punto, nel sistema funzioneranno di nuovo tutti i dischi, garantendoci dal malfunzionamento di uno di essi.

   Per adesso non è possibile assegnare singoli dischi spare a diverse serie RAID. Ogni sottosistema richiederà il proprio disco spare.

10. **D**: Vorrei che ci fosse un allarme udibile per “ehi tu, si è rotto un disco nel mirror” in modo tale che anche il novello amministratore di sistema possa sapere che c’è un problema.

    R: Un evento come questo viene segnalato nei log di sistema dal kernel con la priorità “`KERN_ALERT`”. Vi sono diversi pacchetti software che controllano i file log di sistema e fanno emettere un bip allo speaker del PC, eseguono un pager, mandano e-mail ecc. automaticamente.

11. **D**: Come posso fare a far partire RAID-5 in modalità degradata (con un disco rotto e non ancora rimpiazzato)?

    R: Gadi Oxman < *gadio@netvision.net.il* > ha scritto: Normalmente, per far funzionare un set di n dischi RAID-5 si deve eseguire:

    ```
    mdadd /dev/md0 /dev/disk1 ... /dev/disk(n)
    mdrun -p5 /dev/md0
    ```

    Anche se uno dei dischi è rotto, si deve comunque eseguire `mdadd` su quel disco, come se tutto fosse normale. (?? provare a usare /dev/null al posto del disco rotto ??? occhio)

    Quindi,

    Il sistema RAID funzionerà in modalità degradata con (n - 1) dischi. Se l’esecuzione di “`mdrun`” non va a buon fine, il kernel ha notato un errore (per esempio diversi dischi rotti o uno shutdown sporco). Si usi “`dmesg`” per visionare i messaggi di errore del kernel generati dall’esecuzione di “`mdrun`”. Se il set raid-5 è rovinato a causa di un calo di tensione, e non a causa della rottura di un disco, si può tentare di recuperare il tutto creando un nuovo superblock RAID:

    ```
    mkraid -f --only-superblock raid5.conf
    ```

    Una serie di dischi RAID non fornisce protezione alcuna contro i difetti dell’alimentazione o contro un blocco del kernel e quindi non può nemmeno garantire un corretto recupero dei dati. La ricostruzione del superblock servirà solo a far ignorare al sistema la condizione in cui si trova marcando tutti i drive come “OK” come se niente fosse accaduto.

12. **D**: Come funziona RAID-5 nel caso che un disco smetta di funzionare?

    R: Uno scenario tipico è il seguente:

    - Una serie di dischi RAID-5 è attiva.
    - Un disco si rompe mentre la serie è attiva.
    - Il firmware del disco e i driver di basso livello del disco/controller si accorgono del malfunzionamento e inviano un messaggio di errore al driver MD.
    - Il driver MD continua a fornire un dispositivo `/dev/md0` privo di errori ai driver di livello più alto (anche se con performance ridotte) usando i rimanenti drive operativi.
    - L’amministratore di sistema può quindi eseguire `umount /dev/md0` e `mdstop /dev/md0` come al solito.
    - Se il disco rotto non viene rimpiazzato, l’amministratore di sistema può comunque far partire il sistema raid in modalità degradata come al solito eseguendo `mdadd` e `mdrun`.

13. **D**: Ho appena sostituito un disco rotto in un sistema RAID-5. Dopo aver ricostruito il sistema, l’esecuzione di `fsck` mi dà molti, molti errori. È normale?

    R: No. E, a meno che fsck non sia stato eseguito in modalità verify only; do not update (verifica solamente; non aggiornare. ndt), è del tutto possibile che abbiate rovinato i vostri dati. Sfortunatamente, uno scenario non poco raro è quello nel quale si cambia accidentalmente l'ordine dei dischi in una serie di dischi RAID-5, dopo averne rimpiazzato uno. Anche se il superblock RAID memorizza l'ordine corretto, non tutti i tool tengono conto di questa informazione. In particolare, la versione attuale di `ckraid` userà le informazioni specificate nell'opzione `-f` (usualmente il file `/etc/raid5.conf`) al posto di quelle contenute nel superblock. Se l'ordine specificato non è quello corretto il disco rimpiazzato sarà ricostruito in maniera errata. Sintomo di questo tipo di sbaglio sembrano essere i pesanti e numerosi errori di `fsck`.

E, nel caso siate meravigliati, **sì**, qualcuno ha perso **tutti** i suoi dati commettendo questo sbaglio. È **fortemente raccomandato** un backup su nastro di **tutti** i dati prima di riconfigurare una serie di dischi RAID.

14. **D**:

    R:

15. **D**: Perché non vi è una domanda numero 13?

    A: Se siete scettici sul RAID, sull'Alta Affidabilità e UPS probabilmente è una buona idea l'essere anche superstiziosi. Male non può fare, no?

16. **D**: Il QuickStart dice che `mdstop` serve solo ad essere sicuri che i dischi siano in sincronia. È REALMENTE necessario? Non è abbastanza smontare il filesystem?

    R: Il comando `mdstop /dev/md0`:

    - lo marcherà come "pulito". Questo ci consente di rilevare gli shutdown sporchi dovuti, per esempio, ad un blocco del kernel o ad un malfunzionamento nell'alimentazione.
    - metterà in sincronia i dischi del dispositivo. Questa è una cosa meno importante dello smontare il filesystem ma è importante se l'accesso a `/dev/md0` avviene direttamente invece che attraverso un filesystem (ad esempio come nel caso di `e2fsck`).

5 Risoluzione dei problemi di installazione

1. **D**: Quale è attualmente la patch più stabile e conosciuta per RAID nei kernel della serie 2.0.x?

    R: Al 28 Settembre 1997 è (riporto letteralmente. ndt) 2.0.30 + pre-9 2.0.31 + Werner Fink's swapping patch + the alpha RAID patch. A Novembre 1997, è 2.0.31 + ... !?

2. **D** Le patch per il RAID non vengono correttamente installate. Dov'è l'errore?

    R: Assicuratevi che `/usr/include/linux` sia un link simbolico a `/usr/src/linux/include/linux`.

    Assicuratevi che i nuovi files `raid5.c`, etc. siano stati copiati nei posti giusti. A volte il comando patch non crea nuovi files. Provate con l'opzione `-f` del comando `patch`.

3. **D**: Durante la compilazione di raidtools 0.42, il compilatore si blocca mentre cerca di includere <pthread.h> ma questo file non esiste nel mio sistema. Come posso correggere questo errore?

    R: raidtools-0.42 richiede linuxthreads-0.6 da:
    `ftp://ftp.inria.fr/INRIA/Projects/cristal/Xavier.Leroy`
    In alternativa si possono usare le glibc v2.0.

4. **D**: Ottengo il messaggio: `mdrun -a /dev/md0: Invalid argument`

R: Si deve usare `mkraid` per inizializzare il set RAID prima che venga usato per la prima volta. `mkraid` si assicura del fatto che il sistema RAID sia inizialmente in uno stato di coerenza cancellando le partizioni RAID. In aggiunta,`mkraid` si occuperà di creare i superblock RAID.

5. **D**: Ottengo il messaggio: `mdrun -a /dev/md0: Invalid argument` La procedura di setup è stata:

   - compilazione di raid come modulo del kernel
   - è stata seguita la normale procedura di installazione ... mdcreate, mdadd, etc.
   - il comando `cat /proc/mdstat` produce questo output:

   ```
   Personalities :
   read_ahead not set
   md0 : inactive sda1 sdb1 6313482 blocks
   md1 : inactive
   md2 : inactive
   md3 : inactive
   ```

   - `mdrun -a` genera il messaggio di errore `/dev/md0: Invalid argument`

   R: Si provi ad eseguire `lsmod` (o, in alternativa, `cat /proc/modules`) per vedere se i moduli raid sono stati caricati. Se non lo sono stati, possono essere caricati in maniera esplicita con i comandi `modprobe raid1` o `modprobe raid5`. In alternativa, se usate l'autoloader e se credete che `kerneld` debba caricarli e non lo fa, potrebbe essere a causa del fatto che il loader manca delle informazioni che servono per caricare i moduli. Modificate `/etc/conf.modules` aggiungendo le linee seguenti:

   ```
   alias md-personality-3 raid1
   alias md-personality-4 raid5
   ```

6. **D**: Durante l'esecuzione del comando `mdadd -a` si ha l'errore: `/dev/md0: No such file or directory`. Sembra però che non vi siano `/dev/md0` da nessuna parte. E adesso?

   R: Il package raid-tools crea questi dispositivi quando viene eseguito il comando `make install` come utente root. In alternativa, si può fare così:

   ```
   cd /dev
   ./MAKEDEV md
   ```

7. **D**: Dopo aver creato un sistema raid su `/dev/md0`, provo a montarlo ma ottengo il seguente errore: `mount: wrong fs type, bad option, bad superblock on /dev/md0, or too many mounted file systems`. Cosa c'è che non va?

   R: Si deve creare un file system su `/dev/md0` prima che sia possibile montarlo. Usare `mke2fs`.

8. **D**: Truxton Fulton ha scritto:

   Sul mio sistema Linux 2.0.30, mentre eseguivo `mkraid` su un dispositivo RAID-1, durante la pulizia delle due distinte partizioni ho visto apparire sulla console gli errori - `Cannot allocate free page` e altri errori `Unable to handle kernel paging request at virtual address ...` risultavano nel log di sistema. A questo punto il sistema è diventato pressoché inutilizzabile, ma si è poi ristabilito dopo un po'. L'operazione sembra essersi conclusa senza errori e adesso utilizzo senza problemi il mio dispositivo RAID-1. Comunque quegli errori continuano a sconcertarmi. Qualche idea?

R: Questo era un bug ben conosciuto nei kernel 2.0.30. È stato corretto nel kernel 2.0.31; in alternativa si può tornare al 2.0.29.

9. **D**: Non riesco ad eseguire `mdrun` su un dispositivo RAID-1, RAID-4 o RAID-5. Se provo ad eseguire `mdrun` su un dispositivo aggiunto con `mdadd` mi viene dato il messaggio "`invalid raid superblock magic`".

   R: Assicurarsi che sia stata seguita la parte della procedura di installazione dove viene utilizzato `mkraid`.

10. **D**: Quando accedo a `/dev/md0` il kernel se ne esce con molti errori tipo `md0: device not running, giving up !` e `I/O error...`. Ho aggiunto con successo i miei dispositivi al dispositivo virtuale.

    R: Per essere utilizzabile un dispositivo deve essere in funzione. Si usi il comando `mdrun -px /dev/md0` dove x è 1 per linear, 0 per RAID-0 o 1 per RAID-1, etc.

11. **D**: Ho creato un dispositivo md lineare con 2 dispositivi. `cat /proc/mdstat` mi dice la grandezza totale del dispositivo ma `df` mi fa vedere solo le dimensioni del primo dispositivo fisico

    R: Si deve eseguire `mkfs` su un nuovo dispositivo md prima di usarlo per la prima volta, in modo tale che il filesystem copra tutto il dispositivo.

12. **D**: Ho configurato `/etc/mdtab` usando mdcreate, ho poi eseguito `mdadd`, `mdrun` e `fsck` sulle mie due partizioni `/dev/mdX`. Prima del reboot sembra tutto a posto. Appena effettuo il reboot `fsck` mi dà errori su tutte e due le partizioni: `fsck.ext2: Attempt to read block from filesystem resulted in short read while trying too open /dev/md0`. Perché?! Come posso fare a correggerlo?!

    R: Durante il processo di boot, le partizioni RAID devono essere messe in funzione prima che vengano controllate da `fsck`. Questo deve essere fatto in uno degli script di boot. In qualche distribuzione `fsck` è eseguito da `/etc/rc.d/rc.S`, in altre è eseguito da `/etc/rc.d/rc.sysinit`. Si modifichino questi file in modo da eseguire `mdadd -ar` *prima* di `fsck -A`. Ancora meglio, suggerisco che venga eseguito `ckraid` se `mdadd` restituisce un codice di errore. Come fare ciò è discusso in maggiore dettaglio nella domanda 14 della sezione "Riparare gli errori". (Qui l'originale inglese sembra incoerente, in quanto la domanda 14 della sezione menzionata non è attinente. La domanda più attinente sembra essere la 7 della sezione Considerazioni sul setup e sull'installazione. ndt)

13. **D**: Quando provo a far funzionare un serie di partizioni più grandi di 4 GB mi viene dato il seguente messaggio: `invalid raid superblock magic`

    R: Questo bug è stato corretto. (Settembre 97) Assicuratevi di avere l'ultima versione del codice RAID.

14. **D**: Quando provo ad eseguire mke2fs su una partizione più grande di 2 GB mi viene dato il messaggio `Warning: could not write 8 blocks in inode table starting at 2097175`

    R: Questo sembra essere un problema con `mke2fs` (Novembre 97) Un rimedio temporaneo consiste nel procurarsi il codice di mke2fs e aggiungere `#undef HAVE_LLSEEK` a `e2fsprogs-1.10/lib/ext2fs/llseek.c` subito prima del primo `#ifdef HAVE_LLSEEK` e quindi ricompilare mke2fs.

15. **D**: `ckraid` non riesce a leggere `/etc/mdtab`

R: Il formato del file di configurazione usato per RAID0/linear in `/etc/mdtab` è obsoleto, anche se sarà supportato ancora per un po'. I file di configurazione usati attualmente sono chiamati `/etc/raid1.conf`, etc.

16. **D**: I moduli delle personality (`raid1.o`) non vengono caricati automaticamente; si deve eseguire modprobe manualmente prima di eseguire mdrun. Come posso ovviare all'inconveniente?

    R: Per il caricamento automatico dei moduli, si possono aggiungere le seguenti linee a `/etc/conf.modules`:

    ```
    alias md-personality-3 raid1
    alias md-personality-4 raid5
    ```

17. **D**: Ho aggiunto con `mdadd` 13 dispositivi e adesso sto cercando di eseguire `mdrun -p5 /dev/md0`, ma mi viene dato il messaggio: `/dev/md0: Invalid argument`

    R: La configurazione predefinita di software RAID prevede 8 dispositivi reali. Editare `linux/md.h`, modificare `#define MAX_REAL=8` con un numero più alto e ricompilare il kernel.

18. **D**: Non riesco a far funzionare `md` su delle partizioni nella nostra ultima SPARCstation 5. Sospetto che sia qualcosa che ha a che fare con le etichette di volume.

    R: Le etichette di volume Sun risiedono nel primo 1K di una partizione. Per RAID-1 le etichette di volume Sun non sono un problema poiché `ext2fs` salterà l'etichetta di ogni mirror. Per gli altri livelli RAID (0, linear e 4/5) questo sembra essere un problema; non si è arrivati ancora ad una causa certa (Dicembre 97).

6 Hardware & Software Supportato

1. **D**: Ho un adattatore SCSI della marca XYZ (con o senza diversi canali) e dischi di marche PQR e LMN, funzioneranno con md per creare una personality linear/stripped/mirrored?

    R: Si! Software RAID funzionerà con ogni controller di dischi (IDE o SCSI) e con ogni disco. Non c'è bisogno che i dischi siano identici e nemmeno i controller. Per esempio un mirror RAID può essere creato una metà con dischi SCSI e l'altra metà con dischi IDE. I dischi non devono neppure essere delle stesse dimensioni. Non ci sono restrizioni nella scelta e disposizione dei dischi e dei controller.

    Questo a causa del fatto che Software RAID lavora con le partizioni e non direttamente con il disco. La sola raccomandazione da fare per il RAID livelli 1 e 5 è che le partizioni dei dischi che dovranno essere usate insieme siano delle stesse dimensioni. Se le partizioni che compongono una serie di dischi RAID 1 o 5 non sono delle stesse dimensioni lo spazio in eccesso nella partizione più grande andrà perduto.

2. **D**: Ho un BT-952 a doppio canale, e la confezione asserisce che supporta il RAID hardware livelli 0, 1 e 0+1. Ho messo su un set RAID con due dischi, la scheda sembra riconoscerli durante l'esecuzione delle routine di avvio del suo BIOS. Ho letto il codice sorgente del driver ma non ho trovato riferimenti ad un supporto per il RAID hardware. Qualcuno ci sta lavorando?

    R: Le schede Mylex/BusLogic FlashPoint con RAIDPlus sono in effetti RAID software e per nulla RAID hardware. RAIDPlus è supportato solo su Windows 95 e Windows NT, non su Netware o su piattaforma Unix. Esclusi il boot e la configurazione la gestione del RAID avviene tramite driver del S.O.

Anche se in teoria il supporto per RAIDPlus sia possibile in Linux, l'implementazione del RAID-0/1/4/5 nel kernel di Linux è molto più flessibile e dovrebbe avere performance superiori, quindi ci sono poche ragioni per decidere di supportare direttamente RAIDPlus.

3. **D**: Voglio installare RAID su un computer SMP. RAID è adatto a SMP?

   R: Penso di si è la migliore risposta disponibile al momento (Aprile 98). Diversi utenti dicono di aver usato RAID con SMP per circa un anno senza problemi. Comunque alla data di Aprile 98 (circa kernel 2.1.9x), i seguenti problemi sono stati notati sulla mailing list:

   - I driver Adaptec AIC7xxx SCSI non sono del tutto adatti a SMP (nota generica: gli adattatori Adaptec hanno una lunga storia di problemi e disfunzionamenti. Anche se sembrano essere gli adattatori SCSI più disponibili, diffusi e a buon mercato gli Adaptec andrebbero evitati. Dopo aver calcolato il tempo perso, le frustrazioni e i dati andati perduti capirete che gli Adaptec sono il più costoso errore che abbiate mai fatto. Detto questo, se avete problemi legati a SMP con il kernel 2.1.88, provate la patch su ftp://ftp.bero-online.ml.org/pub/linux/aic7xxx-5.0.7-linux21.tar.gz Non sono sicuro del fatto che questa patch sia stata inclusa negli ultimi kernel 2.1.x. Per altre informazioni date un'occhiata agli archivi di mail del Marzo 98 su http://www.linuxhq.com/lnxlists/linux-raid/lr_9803_01/ Come sempre, dato il fatto che i kernel sperimentali della serie 2.1.x sono soggetti a continui cambiamenti, i problemi descritti in queste mailing list potrebbero o non potrebbero essere stati risolti nel frattempo. Caveat Emptor.)
   - È stato riferito che IO-APIC con RAID-0 su SMP non funziona su 2.1.90

7 Modificare una installazione preesistente

1. **D**: I dispositivi MD linear sono espandibili? Si può aggiungere un nuovo drive/partizione e vedere così aumentata la capienza del filesystem preesistente?

   D: Miguel de Icaza < *miguel@luthien.nuclecu.unam.mx* > ha scritto:

   > Ho cambiato il codice di ext2fs per renderlo capace di trattare dispositivi multipli, modificando l'assunzione precedente che assegnava ad un filesystem un solo dispositivo.
   >
   > Così quando si vuole espandere un filesystem basta eseguire un'utilità che apporta le modifiche appropriate sul nuovo dispositivo (la partizione extra) e poi basta solo far sapere al sistema di estendere il fs usando il dispositivo specificato.
   >
   > Il filesystem può essere esteso con dei nuovi dispositivi mentre il sistema sta funzionando, senza bisogno di doverlo arrestare (e quando avrò altro tempo potrete rimuovere dei dispositivi da un volume ext2, ancora senza nemmeno dover andare in modalità single-user o fare altre cose come questa).
   >
   > Potete procurarvi la patch per il kernel 2.1.x sulla mia web page:
   >
   > `http://www.nuclecu.unam.mx/~miguel/ext2-volume`

2. **D**: Posso aggiungere dei dischi ad un sistema RAID-5?

   R: Attualmente, (Settembre 1997), no, non senza cancellare tutti i dati. Una utilità di conversione che lo permetta ancora non esiste. Il problema è che la struttura e l'effettiva disposizione in un sistema RAID-5 dipende dal numero dei dischi che ne fanno parte.

   Ovviamente si possono aggiungere dei dischi facendo un backup del sistema su nastro, cancellando tutti i dati, creando un nuovo sistema e recuperando i dati dal nastro.

3. **D**: Cosa potrebbe succedere al mio set RAID1/RAID0 se sposto uno dei drive facendo diventare da `/dev/hdb` a `/dev/hdc`?

   A causa dei problemi di grandezza/stupidità con i cablaggi/cabinet devo mettere i miei set RAID sullo stesso controller IDE (`/dev/hda` e `/dev/hdb`). Adesso che ho messo a posto un po' di cose vorrei muovere `/dev/hdb` in `/dev/hdc`.

   Cosa potrebbe succedere se mi limito a cambiare i files `/etc/mdtab` e `/etc/raid1.conf` in modo che riflettano la nuova posizione?

   R: Nel caso del RAID-0/linear, si deve stare attenti nello specificare i drive esattamente nello stesso ordine. Quindi nell'esempio di cui sopra il file di configurazione originale era:

   ```
   mdadd /dev/md0 /dev/hda /dev/hdb
   ```

   E il nuovo file di configurazione *deve* essere

   ```
   mdadd /dev/md0 /dev/hda /dev/hdc
   ```

   Per quanto riguarda RAID-1/4/5, il "numero RAID" del drive viene memorizzato nel suo superblock RAID e quindi l'ordine nel quale vengono indicati i dischi non è importante.

   Il RAID-0/linear non ha un superblock a causa del suo vecchio design e del desiderio di mantenere la compatibilità con questo vecchio design.

4. **D**: Posso convertire un mirror RAID-1 formato da due dischi in una serie di tre dischi RAID-5?

   R: Si. Micheal della BizSystems ha trovato un modo per farlo abilmente e astutamente. Però, come virtualmente tutte le manipolazioni di sistemi RAID una volta che essi contengono dati, è pericoloso e soggetto ad errore umano. **Fate un backup prima di cominciare**.

   ```
   Ipotizzo la seguente configurazione:
   ---------------------------------------------
   dischi
   originale: hda - hdc
   partizioni raid1 hda3 - hdc3
   nome dispositivo raid /dev/md0

   nuovo hda - hdc - hdd
   partizioni raid5 hda3 - hdc3 - hdd3
   nome dispositivo raid: /dev/md1

   Si sostituiscano i nomi dei dischi e delle partizioni in modo
   da riflettere la propria configurazione di sistema.
   Questo vale anche per tutti gli esempi di file di configurazione.
   --------------------------------------------
   FATE UN BACKUP PRIMA DI FARE QUALSIASI ALTRA COSA
   1) ricompilare il kernel per includere sia il supporto raid1 che quello raid5
   2) installare il nuovo kernel e accertarsi che siano presenti le
      personality raid
   3) disabilitare la partizione ridondante sul sistema raid 1. Se questa è
      la partizione di root (la mia lo era) dovrete stare più attenti.

    Fare il reboot del sistema senza mettere in funzione i dispositivi raid
    o fate ripartire il sistema da uno di recupero (i tool raid dovranno essere
    disponibili)
   ```

```
 fate partire raid1 in modalità non ridondante
mdadd -r -p1 /dev/md0 /dev/hda3

4) configurate raid5 con un 'buffo' file di configurazione, si noti che
   non viene nominato hda3 e che hdc3 è ripetuto. Questo serve poiché
   i tool raid non accettano una simile impostazione.
-------------------------------
# configurazione raid-5
raiddev                 /dev/md1
raid-level              5
nr-raid-disks           3
chunk-size              32

# disposizione algoritmo di parità
parity-algorithm        left-symmetric

# dischi spare per ricostruzione a caldo
nr-spare-disks          0

device                  /dev/hdc3
raid-disk               0

device                  /dev/hdc3
raid-disk               1

device                  /dev/hdd3
raid-disk               2
---------------------------------------
 mkraid /etc/raid5.conf
5) attivare il sistema raid5 in modalità non ridondante

mdadd -r -p5 -c32k /dev/md1 /dev/hdc3 /dev/hdd3

6) create un filesystem sul dispositivo raid5

mke2fs -b {blocksize} /dev/md1

la dimensione del blocco raccomandata da alcuni è di 4096 al posto della
predefinita 1024. Questo migliora l'utilizzazione della memoria da parte
del kernel e delle routine raid facendo coincidere la grandezza del blocco
con quella della pagina. Io ho trovato un compromesso su 2048 a causa del
fatto che ho un numero relativamente alto di file piccoli nel mio sistema.

7) montate da qualche parte i due dispositivi raid

mount -t ext2 /dev/md0 mnt0
mount -t ext2 /dev/md1 mnt1

8) spostate i dati
```

```
cp -a mnt0 mnt1

9) verificate che i due set di dati siano identici
10) fermate ambedue i dispositivi raid
11) correggete le informazioni contenute nel file raid5.conf
  cambiate /dev/md1 in /dev/md0
  cambiate il primo disco da leggere in /dev/hda3

12) portare il nuovo sistema in modalità ridondante
 (QUESTO DISTRUGGE LE RIMANENTI INFORMAZIONI raid1)

ckraid --fix /etc/raid5.conf
```

8 Domande sulle performance, sui tool e domande stupide in genere

1. **D**: Ho creato un dispositivo RAID-0 con `/dev/sda2` e `/dev/sda3`. Il dispositivo è molto più lento di una singola partizione. Ma allora md è un ammasso di robaccia?

   R: Per usufruire di un dispositivo RAID-0 che funzioni alla massima velocità, si devono utilizzare partizioni di dischi differenti. Oltretutto, mettendo le due metà di un mirror su di un solo disco non ci si cautela da nessun tipo di malfunzionamento del disco.

2. **D**: Dove è la necessità di avere RAID-linear quando RAID-0 fa le stesse cose con migliore efficienza?

   R: Il fatto che RAID-0 abbia sempre una performance migliore non è cosa ovvia; in effetti, in qualche caso, le cose potrebbero andare peggio. Il filesystem ext2fs distribuisce i file su tutta la partizione, e cerca di mantenere contigui tutti i blocchi di un file, nel tentativo di impedirne la frammentazione. Quindi ext2fs si comporta come se ci fosse una striscia (di dimensioni variabili) per ogni file. Se diversi dischi vengono concatenati in un dispositivo RAID-linear, statisticamente i file verranno distribuiti su ogni disco. Quindi, almeno per ext2fs, RAID-linear si comporta in maniera molto simile a un RAID-0 con delle ampie strisce. Al contrario RAID-0 con strisce piccole può causare un'eccessiva attività del disco che può portare ad un forte degrado delle prestazioni se si accede contemporaneamente a diversi grandi file. In molti casi RAID-0 può risultare facile vincitore. Si immagini, per esempio un grande file di database. Poiché ext2fs cerca di raggruppare insieme tutti i blocchi di un file, vi sono buone possibilità che esso finisca in un solo disco se si utilizza RAID-linear o finisca diviso in molteplici strisce se si usa RAID-0. Si immaginino adesso un certo numero di thread (del kernel) che stanno tentando di accedere al database in maniera casuale. Sotto RAID-linear tutti gli accessi finirebbero con il dover essere soddisfatti da un solo disco che finirebbe con l'essere inefficiente se paragonato alla possibilità di accessi multipli paralleli che RAID-0 consente.

3. **D**: Come si comporta RAID-0 in una situazione nella quale le diverse partizioni di stripe hanno dimensioni diverse? Le strisce vengono distribuite uniformemente?

   R: Per comprendere meglio aiutiamoci con un esempio che coinvolge tre partizioni; una da 50Mb, una da 90Mb e una da 125Mb.

Chiamiamo D0 il disco da 50Mb, D1 il disco da 90Mb e D2 quello da 125Mb. Quando si fa partire il dispositivo, il driver calcola le 'strip zones' (letteralmente zone di striscia. ndt). In questo caso vengono individuate 3 zone, così definite:

```
Z0 : (D0/D1/D2) 3 x 50 = 150MB  totali in questa zona
Z1 : (D1/D2)  2 x 40 = 80MB totali in questa zona
Z2 : (D2) 125-50-40 = 35MB totali in questa zona.
```

Si può notare come la dimensione totale delle zone sia la dimensione del dispositivo virtuale, ma la distribuzione delle strisce varia in funzione della zona. Z2 è inefficiente, poiché contenuta in un solo disco.

Poiché `ext2fs` e molti altri filesystem di Unix distribuiscono i file su tutto il disco, si ha il 35/265 = 13% di probabilità che i dati finiscano su Z2, e quindi non beneficino dello striping.

(DOS cerca di riempire un disco partendo dall'inizio e andando verso la fine e quindi i file più vecchi finirebbero in Z0. Questo tipo di approccio porta però ad una pesante frammentazione, e questo è il perché nessun altro oltre a DOS gestisce il disco in questa maniera).

4. **D**: Ho dei dischi di marca X e un controller di marca Y, sto considerando se usare `md`. Ma il throughput aumenta sensibilmente? Le prestazioni sono notevolmente migliori?

   R: La risposta dipende dalla configurazione che si usa.

   Prestazioni di Linux MD RAID-0 e RAID-linear:
   Se il sistema deve sopperire ad un alto numero di richieste di I/O, statisticamente qualcuna andrà su un disco e qualcun'altra su un altro. Quindi le prestazioni migliorano rispetto ad un singolo disco. Ma il miglioramento effettivo dipende molto dai dati, dalla dimensione delle strisce e da altri fattori. In un sistema con basso carico di I/O le prestazioni sono uguali a quelle di un singolo disco.

   Prestazioni in lettura di Linux MD RAID-1(mirroring):
   MD implementa il bilanciamento in lettura. Quindi il codice RAID-1 distribuirà il carico su ognuno dei dischi nel mirror (due o più), effettuando operazioni alternate di lettura da ognuno di essi. In una situazione con basso carico di I/O questo non influisce per niente sulle prestazioni: dovrete aspettare che un disco abbia finito di leggere. Ma con due dischi in una situazioni di alto carico di I/O la performance in lettura può raddoppiare visto che le letture possono essere effettuate in parallelo da ciascuno dei due dischi. Per N dischi nel mirror, la prestazione può essere N volte migliore.

   Prestazioni in scrittura di Linux MD RAID-1 (mirroring):
   Si deve attendere che la scrittura sia stata effettuata su tutti i dischi del mirror. Questo a causa del fatto che una copia dei dati deve essere scritta su ogni disco del mirror. Quindi le prestazioni saranno quasi uguali a quelle di un singolo disco in scrittura.

   Prestazioni in lettura di Linux MD RAID-4/5:
   Statisticamente un dato blocco può trovarsi in un qualsiasi disco di una serie, e quindi le prestazioni in lettura di RAID-4/5 somigliano molto a quelle di RAID-0. Esse variano in funzione dei dati, della dimensione delle strisce e del tipo di utilizzo. Le prestazioni in lettura non saranno buone quanto quelle di una serie di dischi in mirror.

   Prestazioni in scrittura di Linux MD RAID-4/5:
   Questo sistema è in genere considerevolmente più lento di un disco singolo. Questo a causa del fatto che la parità dovrà essere scritta su un disco e i dati su un altro. E per poter calcolare la nuova parità quella vecchia e i vecchi dati devono prima essere letti.

Viene quindi effettuato un XOR fra i vecchi dati, i nuovi dati e la vecchia parità: questo richiede numerosi cicli di CPU e diversi accessi al disco.

5. **D**: Quale configurazione ottimizza le prestazioni di RAID?

   R: Interessa più massimizzare il throughput o diminuire la latenza? Non vi è una facile risposta dato il grande numero di fattori che influenzano la performance:

   - sistema operativo - l'accesso al disco è effettuato da un solo processo o da più thread?
   - applicazioni - accedono ai dati in maniera sequenziale o in maniera casuale?
   - file system - raggruppa i file o li distribuisce (ext2fs raggruppa insieme i blocchi di un file e distribuisce i file)
   - driver del disco - numero di blocchi di read ahead (è un parametro impostabile)
   - hardware CEC - un drive controller o più?
   - hd controller - gestisce la coda di richieste multiple? Ha una cache?
   - hard drive - dimensioni del buffer della memoria cache – è abbastanza ampia da gestire la quantità e la velocità degli accessi in scrittura di cui si ha bisogno?
   - caratteristiche fisiche del disco - blocchi per cilindro – accedere a blocchi su differenti cilindri porta il disco ad effettuare molte operazioni di seek.

6. **D**: Quale è la configurazione di RAID-5 che ottimizza la performance?

   R: Poiché RAID-5 genera un carico di I/O che è uniformemente distribuito su diversi dischi, le prestazioni migliori si otterranno quando il set RAID viene bilanciato usando drive identici, controller identici e lo stesso (basso) numero di drive su ciascun controller.

   Si noti comunque che l'uso di componenti identici alzerà la probabilità di malfunzionamenti multipli e simultanei dovuti, per esempio a degli sbalzi repentini, al surriscaldamento o a problemi di alimentazione durante un temporale. Questo tipo di rischio può essere ridotto utilizzando dispositivi di marca e modello differenti.

7. **D**: Quale è la dimensione ottimale di un blocco per un sistema RAID-4/5?

   R: Nell'uso dell'implementazione attuale (Novembre 1997) di RAID-4/5 è fortemente raccomandato che il filesystem venga creato con `mke2fs -b 4096` al posto della dimensione predefinita del blocco che è di 1024 byte.

   Questo perché l'attuale implementazione di RAID-5 alloca una pagina di memoria di 4K per ogni blocco del disco; se un blocco del disco fosse grande solo 1K il 75% della memoria allocata da RAID-5 per l'I/O non verrebbe usata. Se la grandezza del blocco del disco è uguale a quella della pagina di memoria il driver può (potenzialmente) usare tutta la pagina. Quindi, su un filesystem con dei blocchi da 4096 invece che da 1024, il driver RAID potrà potenzialmente gestire una coda di richieste di I/O quattro volte più grande senza usare memoria aggiuntiva.

   Nota: le considerazioni precedenti non si applicano ai driver Software RAID-0/1/linear.

   Nota: le considerazioni sulla pagina di memoria da 4K sono da applicare all'architettura Intel x86. Le dimensioni della pagina di memoria su Alpha, Sparc e altre CPU sono differenti; credo che siano 8k su Alpha/Sparc (????). Aggiustate le asserzioni precedenti in maniera da tenerne conto.

   Nota: se il vostro filesystem contiene un grande numero di piccoli file (file più piccoli di 10KBytes), una frazione considerevole di spazio disco andrà perduta. Questo a causa del fatto che la dimensione dello spazio disco allocata dal filesystem è un multiplo della dimensione del blocco. Allocare dei blocchi di grosse dimensioni per dei piccoli file porta chiaramente ad uno spreco di spazio disco; quindi si potrebbe voler continuare ad utilizzare blocchi di piccole dimensioni, avere una una capacità di immagazzinamento maggiore e non preoccuparsi della memoria persa a causa del fato che le dimensioni della pagina e del blocco non combaciano.

Nota: molti sistemi "tipici" non contengono così tanti piccoli file. Comunque, anche se ci fossero centinaia di piccoli file, questo potrebbe portare alla perdita di 10 - 100 MB di spazio disco, che probabilmente è un compromesso accettabile per avere buone prestazioni se si usano hard disk multi-gigabyte.

Nel caso dei news server, ci potrebbero essere decine o centinaia di migliaia di piccoli file. In questi casi i blocchi di dimensioni minori, e quindi una maggiore capacità di immagazzinamento, potrebbero essere più importanti dell'efficienza dello scheduling di I/O.

Nota: esiste un filesystem sperimentale per Linux che memorizza piccoli file e pezzi di file in un solo blocco. Apparentemente questo influisce in maniera positiva sulla performance quando la dimensione media dei file è molto più piccola della dimensione del blocco.

Nota: Le prossime versioni potrebbero implementare dei dispositivi che renderanno obsolete queste discussioni. Comunque sia la loro implementazione è difficoltosa a causa del fatto che la allocazione dinamica a tempo di esecuzione può portare a dei blocchi; l'implementazione attuale effettua una pre-allocazione statica.

8. **D**: Quanto influenza la velocità del mio dispositivo RAID-0, RAID-4 o RAID-5 la grandezza del chunk (grandezza della striscia)?

R: La grandezza del chunk è la quantità di dati contigui nel dispositivo virtuale che sono contigui anche nel dispositivo fisico. In questo HOWTO chunk e striscia sono la stessa cosa: quella che è comunemente chiamata striscia in altre documentazioni su RAID, nelle pagine del manuale di MD è chiamata chunk. Si parla di strisce o chunk solo per RAID 0, 4 e 5 poiché le strisce non vengono utilizzate nel mirroring (RAID-1) e nella semplice concatenazione (RAID-linear). Le dimensioni della striscia influenzano il tempo di latenza (ritardo) nella lettura e nella scrittura, il throughput (larghezza di banda) e la gestione di operazioni indipendenti (l'abilità di provvedere a richieste di I/O simultanee che si accavallano)

Posto che si usino il filesystem ext2fs e le impostazioni attuali del kernel che regolano il read-ahead, le strisce di grosse dimensioni risultano quasi sempre essere una scelta migliore rispetto a quelle di piccole dimensioni, e strisce di dimensioni confrontabili con la grandezza di un quarto di cilindro del disco potrebbero essere ancora migliori. Per capire questa affermazione, consideriamo gli effetti delle strisce grandi su file piccoli, e delle strisce piccole sui file grandi. La dimensione delle strisce non influenza le prestazioni durante la lettura di piccoli file: per una serie di N dischi il file ha 1/N probabilità di essere interamente contenuto in una striscia in uno dei dischi. Quindi sia la larghezza di banda che la latenza in lettura sono comparabili a quelle di un singolo disco. Ipotizzando il fatto che i file piccoli siano distribuiti in maniera statisticamente uniforme nel filesystem (e, se si usa il filesystem ext2fs, questo dovrebbe essere vero) il numero delle letture simultanee sovrapposte può essere circa N volte maggiore, senza collisioni significanti. Al contrario, se vengono utilizzate strisce di dimensioni molto ridotte e un file grande viene letto sequenzialmente, vi sarà un accesso in lettura da ogni disco del sottosistema. Nella lettura di un singolo file di grandi dimensioni, la latenza sarà almeno raddoppiata, e la probabilità che un blocco si trovi molto distaccato dagli altri aumenterà. Si noti comunque ciò che si ottiene: la larghezza di banda può aumentare di al più N volte nella lettura di un singolo file di grandi dimensioni, poiché N dischi lo leggono simultaneamente (se viene usato il read-ahead per mantenere attivi tutti i dischi). Ma vi è anche un effetto secondario controproducente: se tutti i drive sono occupati nella lettura di un singolo, grande file, il tentativo di leggere un secondo, un terzo file allo stesso tempo causerà un grave contenzioso, e degraderà le prestazioni a causa del fatto che gli algoritmi del disco lo porteranno ad effettuare numerosi seek. Quindi, strisce di grosse dimensioni danno quasi sempre i risultati migliori. L'unica eccezione è costituita dalla situazione nella quale si accede ad un singolo file di grandi dimensioni e si richiede la maggiore larghezza di banda possibile e si usa anche un buon algoritmo di read-ahead, in questo caso sarebbero desiderabili strisce di piccole dimensioni.

Si noti che in precedenza questo HOWTO ha raccomandato strisce di piccole dimensioni per i news spool o per altri sistemi con un gran numero di piccoli file. Questo è stato un cattivo consiglio, ed ecco perché: i news spool contengono non solo molti piccoli file ma anche file sommario di grandi dimensioni e grandi directory. Se il file sommario è più grande della striscia, la sua lettura comporterà un accesso su più dischi, rallentando il tutto come se ogni disco effettuasse un seek. Similmente, l'attuale filesystem ext2fs ricerca nelle directory in maniera lineare e sequenziale. Quindi, per trovare un dato file o inode, in media metà della directory verrà letta. Se la directory è distribuita su più strisce (su più dischi), la lettura della directory (per es. a causa del comando ls) potrebbe rallentare notevolmente. Un grazie a Steven A. Reisman < *sar@pressenter.com* > per questa correzione. Steve ha anche aggiunto:

> Ho scoperto che l'uso di una striscia da 256k dà performance molto migliori. Sospetto che la dimensione ottimale sia quella di un cilindro del disco (o forse la dimensione della cache dei settori del disco). Comunque sia, oggi i dischi hanno zone di memorizzazione con un numero di settori variabile (e le cache dei settori variano anche fra differenti modelli). Non c'è un metodo per assicurarsi che le strisce non oltrepassino i confini del cilindro.

I tool accettano le dimensioni delle strisce in KBytes. Conviene specificare un multiplo della dimensione della pagina per la CPU che si usa (4KB su x86).

9. **D**: Quale è il corretto fattore di stride da usare nella creazione di un filesystem ext2fs sulla partizione RAID? Per stride intendo l'opzione -R nel comando `mke2fs`:

```
mke2fs -b 4096 -R stride=nnn  ...
```

Cosa devo mettere al posto di nnn?

R: L'opzione `-R stride` viene usata per comunicare al filesystem le dimensioni delle strisce RAID. Poiché solo RAID-0,4 e 5 usano le strisce, e RAID-1 (mirroring) e RAID-linear non le usano, questa opzione ha senso solo per RAID-0,4,5.

La conoscenza delle dimensioni delle strisce consente a `mke2fs` di dimensionare i blocchi e i bitmap degli inode in modo tale che non vengano a trovarsi tutti sullo stesso dispositivo fisico. Uno sconosciuto ha contribuito alla discussione scrivendo:

> L'ultima primavera ho notato che in una coppia di dischi uno aveva sempre un I/O maggiore e ho attribuito la cosa a questi blocchi di meta-dati. Ted ha aggiunto l'opzione `-R stride=` in risposta alle mie spiegazioni e alla richiesta di una soluzione.

Per un filesystem con blocchi da 4Kb e strisce da 256Kb, si potrebbe usare `-R stride=64`.

Se non volete affidarvi all'opzione `-R`, potete ottenere un effetto simile in modo differente. Steven A. Reisman < *sar@pressenter.com* > scrive:

> Un'altra questione è l'uso del filesystem su un dispositivo RAID-0. Il filesystem ext2 alloca 8192 blocchi per ogni gruppo. Ogni gruppo ha il proprio set di inode. Se ci sono 2, 4, o 8 dischi questi blocchi si accumulano nel primo disco. Ho distribuito gli inode su tutti i drive impostando mke2fs in modo da allocare solo 7932 blocchi per gruppo.

Qualche pagina di mke2fs non descrive l'opzione `[-g blocks-per-group]` usata in questa operazione

10. **D**: Dove posso mettere i comandi `md` negli script di avvio, in modo tale che tutto parta automaticamente al boot?

R: Rod Wilkens < *rwilkens@border.net* > scrive:

> Quello che ho fatto è stato mettere "`mdadd -ar`" nel "`/etc/rc.d/rc.sysinit`" subito dopo il punto nel quale il kernel carica i moduli, e prima del controllo dischi di "`fsck`". In questa maniera si può mettere il dispositivo "`/dev/md?`" in "`/etc/fstab`". Quindi ho messo il comando "`mdstop -a`" subito dopo il comando "`umount -a`" nel file "`/etc/rc.d/init.d/halt`".

Nel caso si usi raid-5 si dovrà fare attenzione al codice di uscita di `mdadd` e, nel caso indichi un errore, eseguire

```
ckraid --fix /etc/raid5.conf
```

per riparare i danni.

11. **D**: Mi chiedo se sia possibile configurare lo striping su più di 2 dispositivi in `md0`? Questo per un news server, e io ho 9 dischi... Non c'è bisogno che dica che ne servono molti più di due. È possibile?

    A: Si. (descrivere come)

12. **D**: Quando Software RAID è superiore al RAID Hardware?

    R: Normalmente il RAID hardware è considerato superiore al RAID Software, poiché i controller hardware dispongono spesso di una capiente cache e possono effettuare una programmazione migliore delle operazioni in parallelo. Comunque il software RAID integrato può (e lo fa) avvantaggiarsi della sua integrazione con il sistema operativo.

    Per esempio, ... ummm. Oscura descrizione del caching dei blocchi ricostruiti nella cache del buffer tralasciata ...

    È stato riferito che, su un sistema SMP con doppio PPro, software RAID supera le prestazioni di un hardware RAID di ben nota marca di un fattore variabile da 2 a 5.

    Software RAID è anche un'opzione molto interessante per sistemi server ridondanti ad altro gradi di affidabilità. In questa configurazione due CPU sono collegate ad un set di dischi SCSI. Se un server si blocca o non risponde più l'altro server può eseguire `mdadd`, `mdrun` e `mount` per montare la serie di dischi RAID, e continuare le operazioni. Questo tipo di operazione a doppio controllo non è sempre possibile con molti controller RAID, a causa del fatto che il controller hardware mantiene la stessa configurazione.

13. **D**: Se aggiorno la mia versione di raidtools, posso avere problemi nella gestione di vecchi sistemi? In breve, devo ricreare i miei sistemi RAID ogni volta che aggiorno i programmi di utilità raid?

    R: No, a meno che non cambi il numero primario di versione. Una versione di MD x.y.z consiste di tre sottoversioni:

    ```
    x:      Versione primaria.
    y:      Versione secondaria.
    z:      Livello di patch.
    ```

    La versione x1.y1.z1 del driver RAID supporta un sistema RAID con versione x2.y2.z2 nel caso (x1 == x2) e (y1 >= y2).

    Le versioni che differiscono per il solo livello di patch (z) sono concepite in modo da essere compatibili.

    Il numero di versione secondario viene incrementato quando la struttura del sistema RAID viene modificata in modo tale da renderla incompatibile con le vecchie versioni del driver. Le nuove versioni del driver manterranno la compatibilità con i vecchi sistemi RAID.

    Il numero primario di versione viene incrementato quando non vi sono più ragioni per continuare a supportare i vecchi sistemi RAID nel nuovo codice del kernel.

Per quanto riguarda RAID-1, è improbabile che la struttura del disco o dei superblock venga alterata entro breve termine. Le ottimizzazioni e le nuove funzioni (ricostruzione, tool che implementino il multithread, hot-plug ecc.) non vanno a modificare la struttura fisica.

14. **D**: Il comando `mdstop /dev/md0` dice che il dispositivo è occupato.

    R: C'è un processo che ha un file aperto su `/dev/md0` o `/dev/md0` è ancora montato. Chiudere il processo o eseguire `umount /dev/md0`.

15. **D**: Vi sono dei tool per l'analisi delle prestazioni?

    R: Vi è anche un nuovo programma di utilità chiamato `iotrace` nella directory `linux/iotrace`. Esso legge `/proc/io-trace` e analizza/riporta il suo output. Se credete che le prestazioni dei vostri dispositivi a blocchi siano poco convincenti, date un'occhiata all'output di iotrace.

16. **D**: Leggendo i sorgenti di RAID ho visto il valore `SPEED_LIMIT` impostato a 1024K/sec. Che significa? Questo rallenta le prestazioni?

    R: `SPEED_LIMIT` viene usato per regolare la ricostruzione RAID quando essa avviene in automatico. Semplificando, la ricostruzione automatica permette di effettuare `e2fsck` e `mount` subito dopo uno shutdown sporco, senza prima dover eseguire `ckraid`. La ricostruzione automatica viene usata anche dopo la sostituzione di un disco rotto.

    Per evitare un sovraccarico del sistema mentre la ricostruzione è in corso, il processo di ricostruzione controlla la velocità alla quale essa avviene e la rallenta se è troppo veloce. Il limite di 1M/sec è stato scelto arbitrariamente come ragionevole velocità che consente alla ricostruzione di finire in un tempo accettabile, con solo un leggero carico del sistema, in modo tale che gli altri processi non vengano disturbati.

17. **D**: E riguardo la "spindle synchronization" o "disk synchronization" (sincronizzazione dei dischi. ndt)?

    R: La sincronizzazione dei dischi viene usata per far girare più hard disk esattamente alla stessa velocità, in modo tale che le loro superfici siano sempre perfettamente allineate. Questo metodo viene usato da qualche controller hardware per migliorare l'organizzazione degli accessi in scrittura. Tuttavia, per quanto riguarda Software RAID, questa informazione non viene usata e la sincronizzazione dei dischi può addirittura influire negativamente sulle prestazioni.

18. **D**: Come posso creare degli spazi di swap usando raid 0? Lo stripe delle aree di swap su più di 4 dischi è realmente veloce?

    R: Leonard N. Zubkoff risponde: È veramente veloce, ma non c'è necessità di usare MD per mettere in strip le aree di swap. Il kernel usa automaticamente le strisce su diverse aree di swap a priorità uguale. Per esempio, la seguente configurazione di `/etc/fstab` mette in stripe le aree di swap su cinque drive suddivisi in tre gruppi:

    ```
    /dev/sdg1       swap    swap    pri=3
    /dev/sdk1       swap    swap    pri=3
    /dev/sdd1       swap    swap    pri=3
    /dev/sdh1       swap    swap    pri=3
    /dev/sdl1       swap    swap    pri=3
    /dev/sdg2       swap    swap    pri=2
    /dev/sdk2       swap    swap    pri=2
    /dev/sdd2       swap    swap    pri=2
    /dev/sdh2       swap    swap    pri=2
    ```

```
/dev/sdl2       swap    swap    pri=2
/dev/sdg3       swap    swap    pri=1
/dev/sdk3       swap    swap    pri=1
/dev/sdd3       swap    swap    pri=1
/dev/sdh3       swap    swap    pri=1
/dev/sdl3       swap    swap    pri=1
```

19. **D**: Voglio ottimizzare le prestazioni. Devo usare controller multipli?

    R: In molti casi la risposta è si. L'uso di controller multipli per accedere in parallelo al disco consentirà un incremento delle prestazioni. Ovviamente il miglioramento effettivo dipenderà dalla vostra particolare configurazione. Per esempio è stato riferito (Vaughan Pratt, gennaio 98) che un singolo Cheetah da 4.3Gb collegato ad un Adaptec 2940UW può arrivare ad un trasferimento di 14Mb/sec (senza l'uso di RAID). Installando due dischi su un controller e usando una configurazione RAID-0 si arriva ad una prestazione di 27Mb/sec.

    Si noti che il controller 2940UW è un controller SCSI Ultra-Wide, capace di un trasferimento teorico di 40Mb/sec. quindi la velocità di trasferimento misurata non sorprende. Tuttavia un controller più lento collegato a due dischi veloci potrebbe fare da collo di bottiglia. Si noti anche che molte periferiche SCSI out-board (ad es. i tipi con le connessioni utilizzabili a caldo) non possono arrivare a 40Mb/sec a causa del rumore elettrico e di quello dovuto al cablaggio.

    Se state progettando un sistema a controller multipli tenete a mente il fatto che molti dischi e molti controller funzionano normalmente al 70-85% della loro velocità massima.

    Si noti anche che l'uso di un controller per disco può ridurre la probabilità che il sistema si blocchi a causa di un malfunzionamento dei cavi o del controller (Teoricamente – questo accade solo nel caso in cui il driver del controller riesca a gestire ordinatamente un controller rotto. Non tutti i device driver SCSI sembrano riuscire a gestire una simile situazione senza andare in panico o bloccarsi in altra maniera).

9 RAID ad Alta Affidabilità

1. **D**: RAID mi aiuta a cautelarmi dalla perdita di dati. Ma come posso anche assicurarmi che il sistema resti funzionante il maggior tempo possibile, e non sia soggetto a dei blocchi? Idealmente, vorrei un sistema che funzioni 24 ore al giorno, 7 giorni alla settimana, 365 giorni all'anno.

    R: Raggiungere l'Alta Affidabilità è difficile e dispendioso. Più si cerca di rendere indipendente dai guasti il sistema, più il sistema diventa difficile e costoso. I seguenti trucchi, consigli, idee e voci non confermate forse possono aiutarvi nella vostra ricerca.

    - I dischi IDE si possono rompere in maniera tale che il disco rotto su un cavo IDE può impedire al disco funzionante sullo stesso cavo di rispondere, facendo apparire la cosa come se tutti e due i dischi fossero rotti. Poiché RAID non protegge dal malfunzionamento di due dischi si dovrà o mettere un solo disco su un cavo IDE o, se ci sono due dischi sullo stesso cavo, essi devono fare parte di set RAID differenti.
    - I dischi SCSI si possono rompere in maniera tale da impedire l'accesso alle altre unità SCSI collegate in cascata. La modalità di malfunzionamento implica un cortocircuito del pin (condiviso dalle altre unità) attraverso il quale viene segnalato che il dispositivo è pronto; poiché questo collegamento è condiviso, non si possono effettuare operazioni finché il cortocircito non sia rimosso. Quindi, due dischi SCSI sulla stessa catena non devono appartenere allo stesso set RAID.
    - Simili considerazioni valgono anche per i controller. Non usate tutti i canali di un controller; usate diversi controller.

- Non usate la stessa marca o modello per tutti i dischi. Non è improbabile che dei forti temporali ve ne possano rompere due o più (sì, tutti usiamo degli stabilizzatori di corrente, ma questi non sono macchine perfette). Il caldo e l'insufficiente ventilazione del disco sono altri killer di dischi. I dischi a buon prezzo spesso si surriscaldano. L'uso di modelli differenti di dischi e controller diminuisce la probabilità che qualsiasi cosa succeda ad un disco (il caldo, uno shock fisico, vibrazioni, sovratensioni) possa succedere anche agli altri nello stesso modo.
- Per cautelarsi da malfunzionamenti del controller o del PC, potrebbe essere possibile costruire un set di dischi SCSI che sia twin-tailed; collegato cioé a due computer. Un computer monta il filesystem in lettura-scrittura, mentre il secondo computer lo monta in sola lettura, e agisce da hot spare (ricambio a caldo ndt). Quando il computer che agisce da hot spare viene informato del fatto che il computer principale si è rotto (ad es. attraverso un watchdog), toglie tensione al computer principale (per essere sicuri che sia realmente spento) e quindi effettua un fsck e rimonta il filesystem in lettura-scrittura. Se qualcuno riesce a far funzionare questa configurazione lo prego di farmi sapere.
- Usare sempre un UPS ed effettuare shutdown puliti. Anche se uno shutdown sporco può non danneggiare i dischi, l'esecuzione di ckraid su un sistema di dischi anche piccolo è estremamente lenta. Oppure potete hackerare il kernel e fare un debug del codice che riguarda la ricostruzione a caldo...
- I cavi SCSI sono famosi per essere delle creature dal comportamento variabile, soggette ad ogni sorta di accidenti. Usate i migliori cavi che possiate rimediare. Si usi il bubble-wrap per assicurarsi che i cavi non stiano troppo vicino l'uno all'altro generando mutue interferenze. Si osservino rigorosamente le restrizioni sulla lunghezza dei cavi.
- Date un'occhiata a SSI (Serial Storage Architecture). Anche se dispendiosa, si dice che sia più affidabile della tecnologia SCSI.
- Divertitevi, è più tardi di quanto immaginiate.

10 Domande che attendono risposta

1. **D**: Se, per ragioni di prezzo, metto in mirror un disco lento con uno veloce, il software sarà abbastanza scaltro da bilanciare le richieste di lettura tenendo conto della velocità dei dischi o farà rallentare il tutto alla velocità del disco più lento?

2. **D**: Per testare il thru-put del disco... c'è un dispositivo a caratteri cui si possa accedere direttamente al posto di `/dev/sdaxx` che si possa usare per valutare le prestazioni dei dischi raid?? c'è un programma GUI che si possa usare per controllare il thru-put del disco??

11 Desiderata di MD e del relativo software

Bradley Ward Allen < *ulmo@Q.Net* > ha scritto:

> Le idee includono:
>
> - Parametri di boot per dire al kernel quali dispositivi dovranno essere dispositivi MD (niente più "`mdadd`")
> - Rendere MD trasparente a "`mount`"/"`umount`" in modo tale che non vi siano più "`mdrun`" e "`mdstop`"
> - Completa integrazione nel kernel di "`ckraid`" e sua esecuzione automatica in caso di bisogno.

(Ho già suggerito di smetterla di usare i tool e di integrarli nel kernel; io la penso così, si parla di un filesystem, non di un giocattolo.)

- Trattare sistemi che possano facilmente sopravvivere al malfunzionamento (simultaneo o in momenti separati) di N dischi, con N intero > 0 definito dall'amministratore di sistema.
- Migliorarne il comportamento in caso di blocco del kernel, problemi con l'alimentazione e altri shutdown improvvisi.
- Non disabilitare l'intero disco se solo una parte di esso si è rovinata, ad es. se gli errori di lettura sono meno del 50% su 20 diverse richieste di accesso, si continua ad usare il disco ignorando i settori che hanno dato problemi.
- Settori danneggiati:
  - Un meccanismo che consenta di memorizzare da qualche parte nel disco quali settori sono danneggiati.
  - Se esiste già una convenzione riconoscibile dai filesystem di livello più alto per marcare i settori danneggiati, questa deve essere usata. Programmarne una se non ne esiste una riconoscibile.
  - Forse in alternativa un meccanismo per fare sapere allo strato superiore che le dimensioni del disco si sono ridotte, magari implementando una automazione che consenta allo strato superiore di spostare i dati dalle aree che vengono eliminate. Questo potrebbe anche andare bene per trattare i blocchi danneggiati.
  - Nel caso non si possano realizzare le idee di cui sopra, lasciare una piccola parte del disco (definibile dall'amministratore di sistema) da parte per i blocchi danneggiati (magari distribuita su tutto il disco?) e usare questa area (la più vicina) al posto dei blocchi danneggiati quando questi vengono scoperti. Ovviamente questa soluzione è inefficiente. Oltretutto il kernel dovrebbe mettere nei log, ogni volta che il sistema RAID viene avviato, tutti i settori danneggiati e i provvedimenti adottati nei loro riguardi con priorità "`crit`", solo per far sapere all'amministratore che il suo disco è impolverato internamente (o ha una testina malata).
- Dischi (dis)attivabili via software:

  "disattiva questo disco"
  : si blocca fino a che il kernel non si è assicurato che non vi siano dati che possono servire sul disco che sta per essere disattivato (ad es. per completare uno XOR/ECC/ o altra correzione di errore), quindi cessa l'utilizzo del disco (in modo che possa essere rimosso, ecc.)

  "attiva questo disco"
  : esegue, se necessario, `mkraid` su un nuovo disco e quindi lo utilizza per le operazioni ECC/qualsiasi, ampliando quindi il sistema RAID5;

  "ridimensiona il sistema"
  : reimposta il numero totale di dischi e il numero di dischi ridondanti, spesso con il risultato di aumentare le dimensioni del sistema RAID; sarebbe bello poter usare questa opzione, quando serve, senza perdere dati, ma mi viene difficile immaginare come possa funzionare effettivamente; in ogni caso, un modo per sospendere (possibilmente per delle ore (il kernel dovrebbe scrivere qualcosa nei log ogni dieci secondi in questo caso)) potrebbe essere necessario;

  "attiva questo disco mentre salvi i dati"
  : che salvi i dati su un disco così com'è e lo inserisca in un sistema RAID5, in modo tale che l'orrendo salva e ripristina non debba essere eseguito ogni volta che qualcuno configuri un sistema RAID5 (oppure, potrebbe essere più semplice salvare una partizione al posto di due, potrebbe addirittura entrare nella prima come file compresso con gzip); infine,

“riattiva disco”
potrebbe essere un modo grazie al quale l’operatore scavalca il SO per provare un disco che in precedenza era risultato non funzionante (potrebbe semplicemente chiamare disattiva e quindi attiva, penso).

Altre idee dalla rete:

- rendere finalrd simile a initrd, per semplificare il boot da raid.
- una modalità raid di sola scrittura, per rendere più semplice quanto sopra
- Contrassegnare il sistema RAID come pulito quando non siano state effettuate mezze scritture. – Sarebbe come dire che non vi sono operazioni di scrittura finite su un disco e ancora da ultimare su un altro disco.

  Aggiungere un timeout che segnali inattività in scrittura (per evitare seek frequenti al superblock RAID quando il sistema RAID è relativamente occupato.)