POLITECNICO DI MILANO

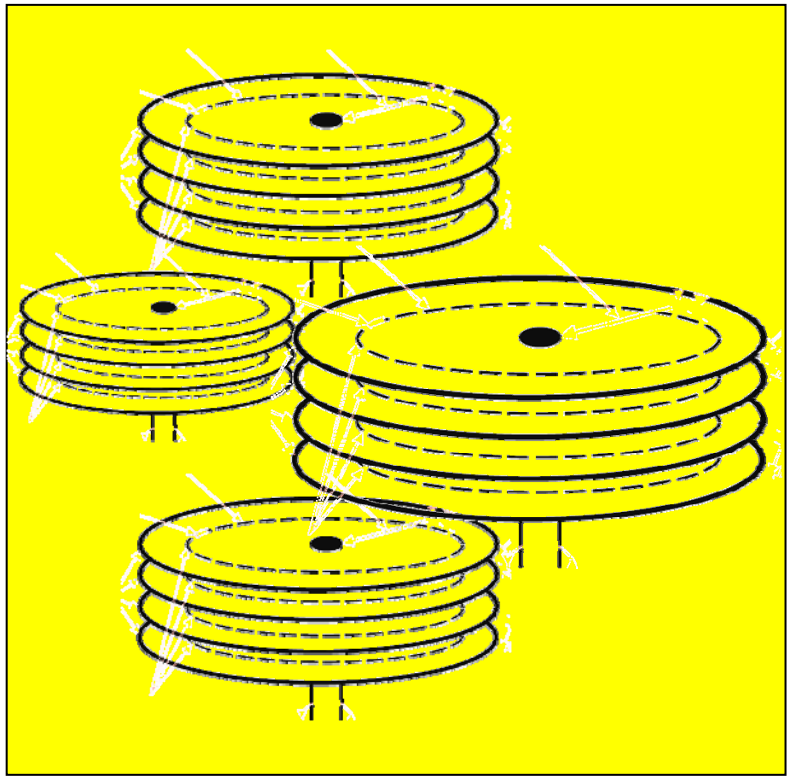
Impianti Informatici





RAID

Redundant Arrays of Independent (Inexpensive) Disks



Parallelismo

Elevato Transfer Rate

- I/O pesanti

Elevato I/O Rate

- I/O leggeri

Load balancing

"A case for Redundant Arrays of Inexpensive Disks (RAID)" - D. Patterson,1988

RAID: Hw vs. Sw

Possono essere realizzati sia con hardware dedicato sia con software che usa hardware standard (esistono anche soluzioni ibride)

- *Soluzioni software* richiedono costi di CPU (cicli aggiuntivi)
- *Soluzioni hardware*
  - richiedono unità di controllo speciali che eseguono i calcoli di parità
  - hanno in genere velocità maggiore delle soluzioni sw. Dipende da:
    - dimensione della cache
    - quanto rapidamente i dati vengono scaricati sui dischi
  - supportano *hot swapping* (se possibile)

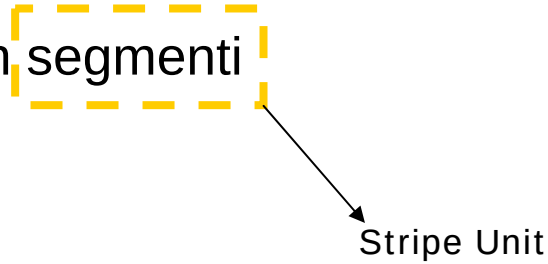
Data Striping

Striping

Aumenta le prestazioni

Distribuzione dei dati su multipli dischi

- In modo trasparente
- I dati sequenziali vengono suddivisi in segmenti
- Scritti con un algoritmo di *round robin*



Stripe Width

- Numero di dischi usati dallo striping
- Può non coincidere con il numero di dischi totali

Ridondanza

Aumenta l'affidabilità

Data striping: prestazioni

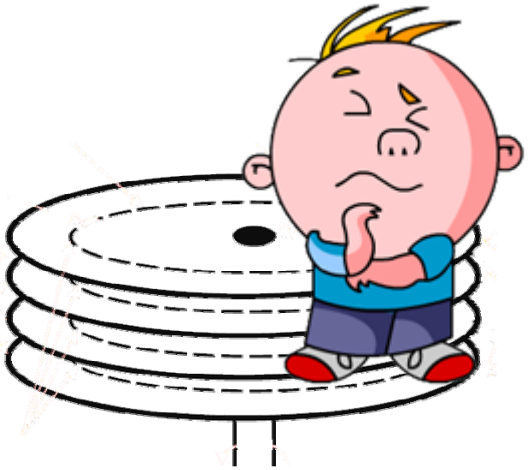
Parallelismo

- Più richieste contemporanee servite in parallelo
  - Si riduce il *queueing time*

- Una singola richiesta di I/O per *multiple block* può essere servita in parallelo da più dischi
  - Aumenta il transfer rate

Vulnerabilità

$P_{guasto}[100\ dischi] = \sim 100 * P_{guasto}[1\ disco]$



MTTF(1 disco) = 200000 = ~23 anni

MTTF(100 dischi) = 2000 = ~3 mesi

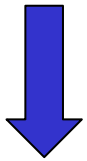
Ridondanza → Dati ridondanti memorizzati su altri dischi

Correzione errori

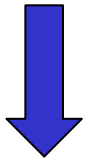
Recupero dati persi

Ridondanza

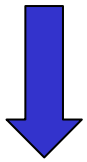
Molteplici dischi



Aumentata vulnerabilità



Scrittura di informazioni ridondanti



Lentezza delle *write*

(Aggiornamento dei dati ridondanti)

Data Interleaving

Data Interleaving

Fine grained

Coarse grained

Stripe Unit

Performance

Unità di stripe

La dimensione influenza le prestazioni

Ad. es.: stripe unit piccola???

SVANTAGGI

Le singole richieste si distribuiscono su più dischi

Riduce l'efficacia di tecniche quali il *prefetching* dei dati

VANTAGGI

Evita dischi con utilizzo asimmetrico (*access skew*)

Data Interleaving: fine grained

Dati scomposti in *stripe unit* piccole

Tutte le richieste di I/O possono usare l'intero array

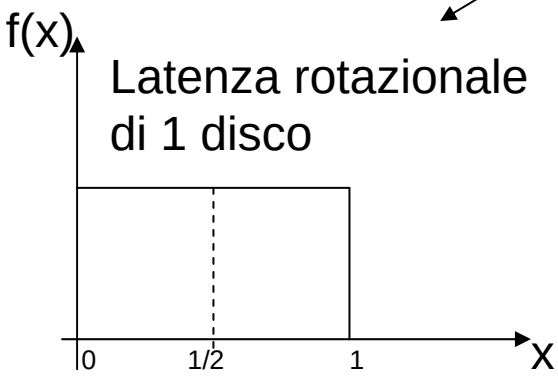
VANTAGGI

Elevato transfer rate

SVANTAGGI

Serve una singola richiesta logica di I/O alla volta

Attesa per il posizionamento di ciascun disco



Ad.es.:
- con un disco E[X1] = ½ giro
-con due dischi E[max(X1,X2)]=7/12giro
-…

Data Interleaving: coarse grained

Dati scomposti in *stripe unit* grandi

- Richieste di I/O *piccole* → Usano pochi dischi
- Richieste di I/O *grandi* → Usano tutti i dischi

Molte richieste di I/O *piccole* servite in parallelo

- Vengono servite più richieste logiche contemporaneamente

Richieste di I/O *grandi* con elevato transfer rate

- Accesso contemporaneo a multipli dischi

Ottimizzazione delle prestazioni

Read Caching

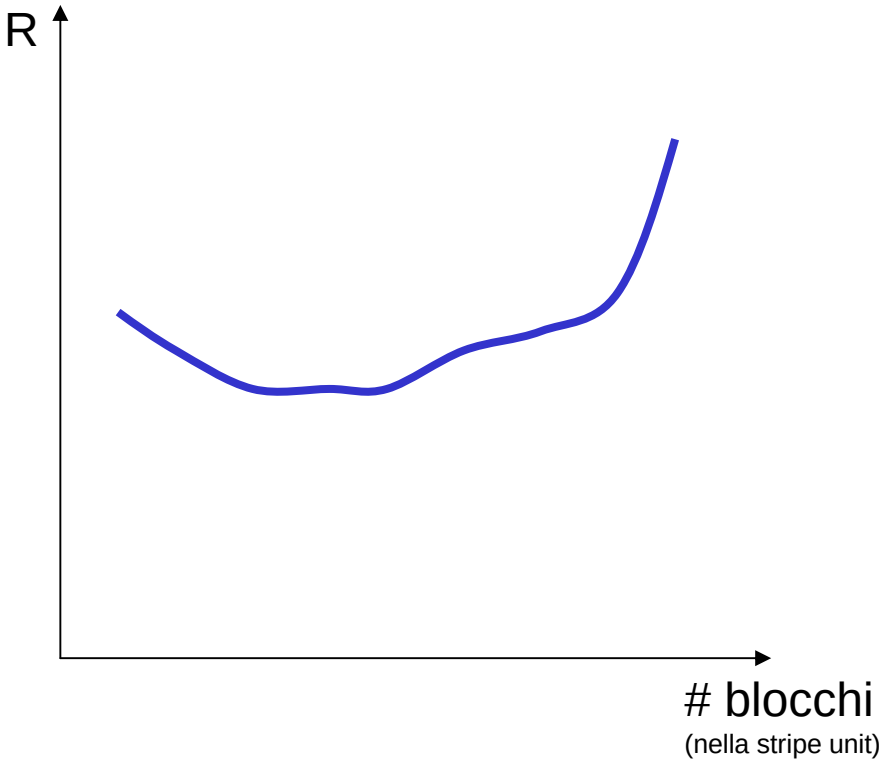
Prefetching

Write Buffering

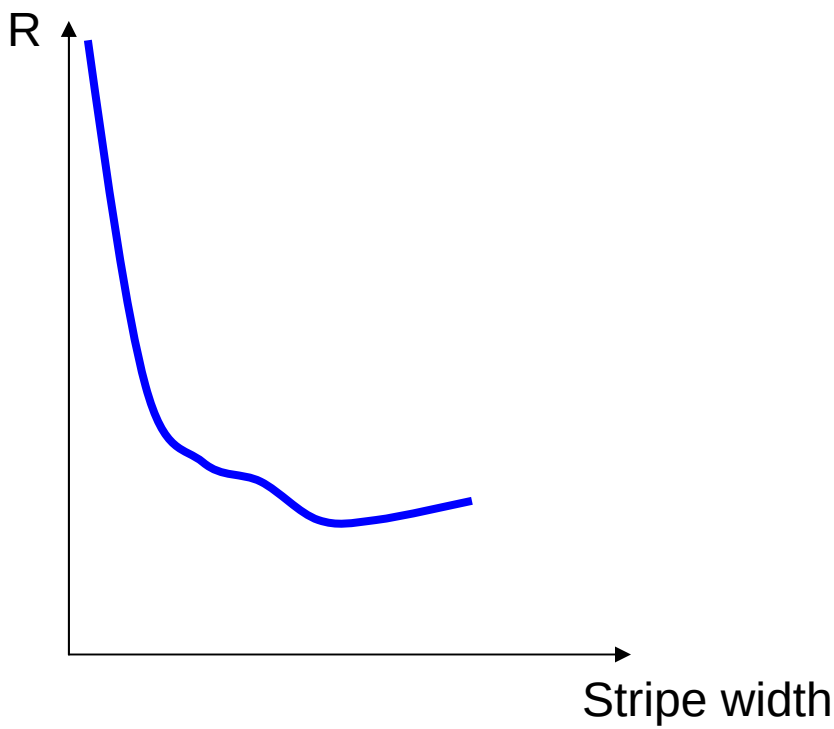
- Quando i dati sono suddivisi su più dischi (*striping*) la *locality* è modificata
  - le regioni attive possono essere contigue su più dischi così che i bracci di posizionamento delle testine hanno una estensione di movimento più limitata
  - lo stesso carico utilizza su ognuno dei diversi dischi solo una frazione dello spazio che userebbe su uno.

Locality in presenza di Striping

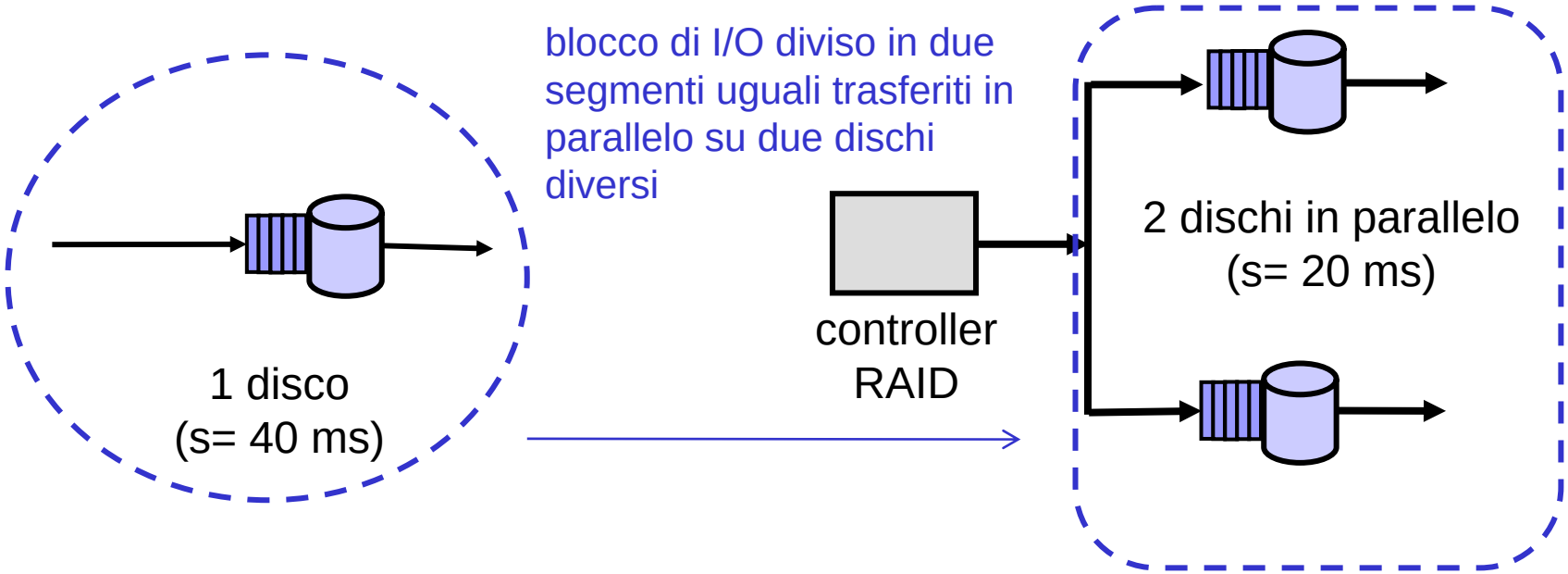
Resp. time in funzione della dimensione della stripe unit (numero di blocchi di una stripe) parità di no. dischi



Resp. time in funzione del numero di dischi a parità di blocchi di stripe



esempio: 2 dischi in parallelo



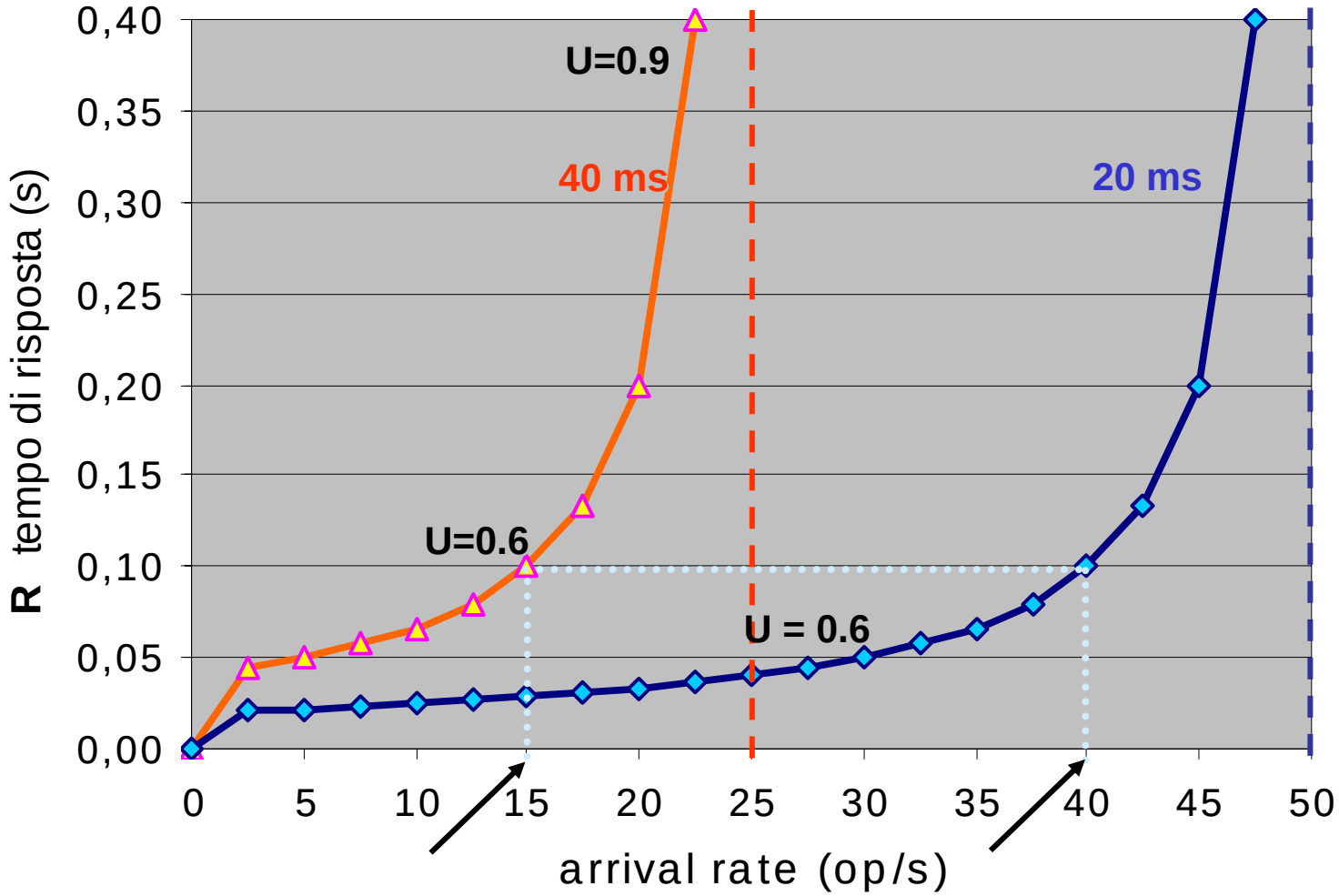
si **ipotizza** che le operazioni vengano effettuate in parallelo sui due dischi con tempi di servizio pari alla metà di quello originario

a *parità di utilizzo* passando da uno a due dischi in parallelo, il tempo richiesto da una singola operazione di I/O si dimezza e il carico si raddoppia (utilizzo 60%, tempi di risposta da 100 ms a 50 ms).

a *parità di tempo di risposta* il carico di lavoro servito cresce (per esempio aumenta del 166% da 15 a 40 operazioni/s con 100 ms di risposta

esempio: calcolo del tempo di risposta

Andamento dei tempi di risposta al variare del carico (operazioni/sec)
per tempi di servizio di 40 e 20 ms



striping: ripartizione uniforme delle operazioni

- Un risultato importante, dal punto di vista delle prestazioni, della tecnica di striping (che in prima approssimazione può essere pensata come la suddivisione di una singola operazione in diverse eseguite in parallelo) è il fatto che gli accessi si ripartiscono automaticamente in modo uniforme fra i dischi interessati
- In condizioni stazionarie una distribuzione omogenea delle richieste fra dispositivi (di identiche caratteristiche) è quella che garantisce il minore tempo medio di risposta

- si veda l'esempio seguente

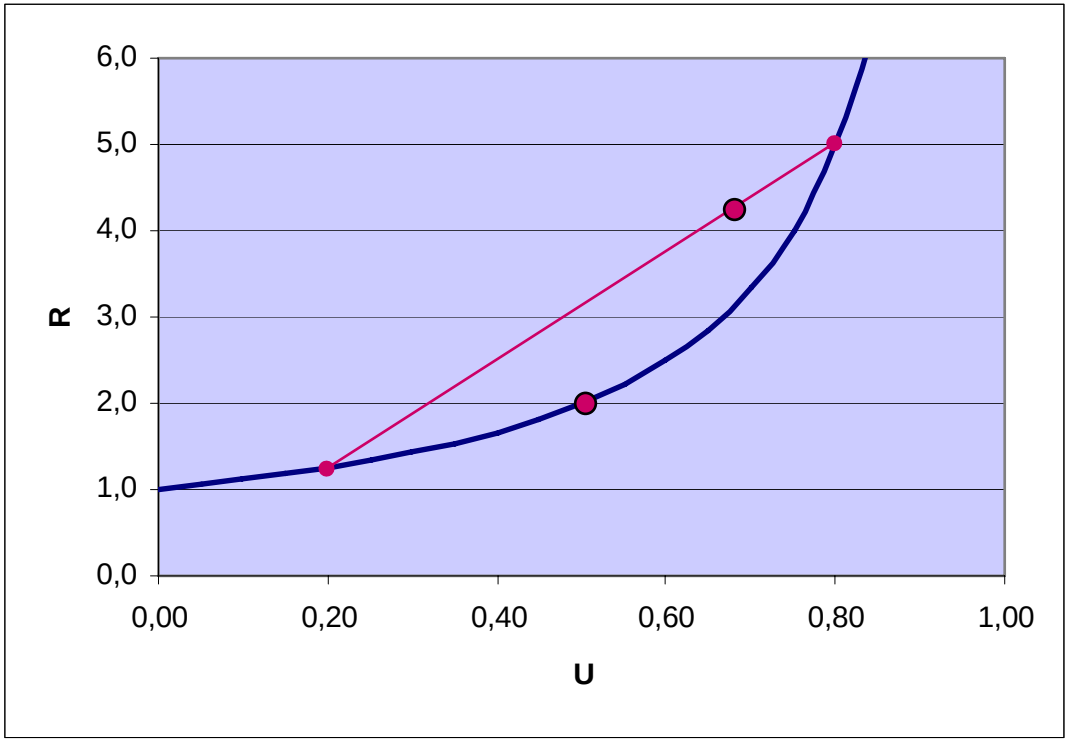
esercizio: ripartizione uniforme delle operazioni

un certo carico può ripartirsi su due dischi secondo due distinte modalità:

- **20% e 80%** oppure **50% e 50%**
- ipotizziamo che se tutto il carico avesse a disposizione un solo disco, questo sarebbe utilizzato al 100%, perciò la ripartizione del carico può essere usata anche come utilizzo dei dischi stessi
- i tempi di risposta R (attesa + servizio) in funzione del carico sono riportati nel grafico successivo (approssimazione M/M/1)

esercizio: ripartizione uniforme delle operazioni (2)

due dischi utilizzati rispettivamente al **20** e **80** percento hanno un tempo medio di risposta di **4.25**
se invece sono utilizzati al **50** percento hanno entrambi un tempo di **2.0**



- nel primo caso abbiamo **R(disco1) = 1.25; R(disco2) = 5**
- nel secondo **R(disco1) = R(disco2) = 2**

- il tempo medio di risposta vale, nelle due ipotesi, rispettivamente :

$0.2 \times 1.25 + 0.8 \times 5 =$ **4.25** (ripartizione disomogenea)

$0.5 \times 2 + 0.5 \times 2 =$ **2** (ripartizione omogenea)

Impianti Informatici

POLITECNICO DI MILANO





RAID

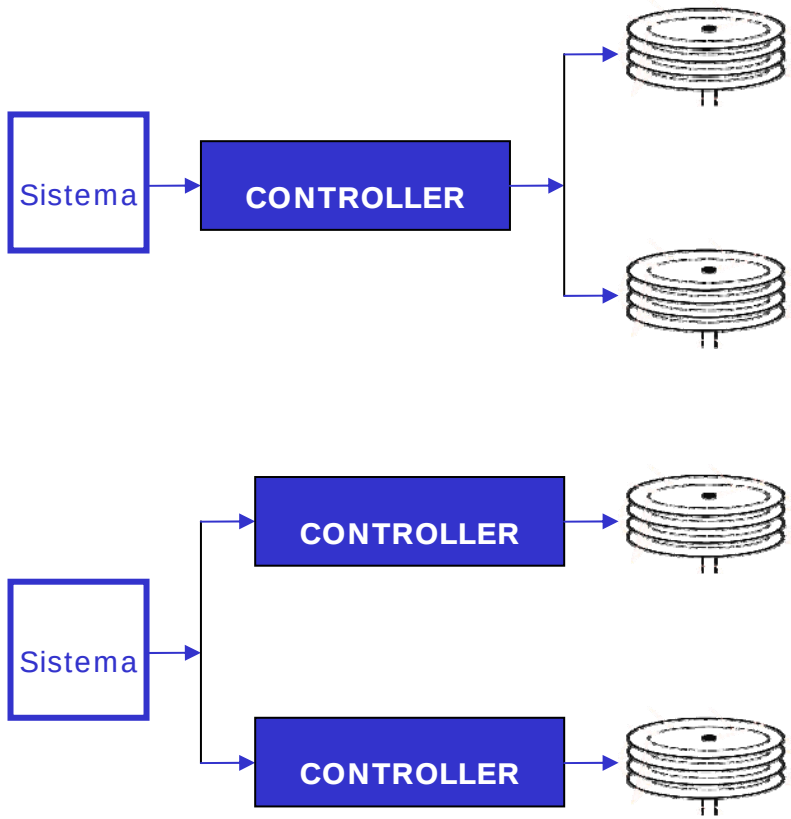
Tecnologie adottate

Mirroring

Duplexing (splitting)

Parity

Striping



I livelli di RAID

Non esiste un unico tipo di RAID

Ci sono molti ***livelli***

- Tecnologia
- Configurazione
- Obiettivi

Il *controller* determina quali livelli possono essere implementati

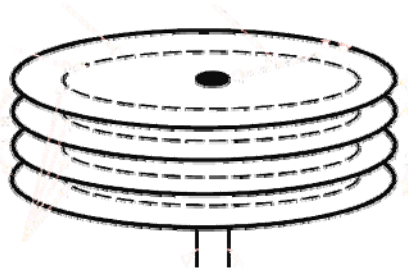
- Per alcuni livelli non è necessario un *controller RAID*
  - Sistema operativo
  - Software di management dell'array

I livelli di RAID

RAID 0

RAID 1

- RAID 0+1
- RAID 1+0



RAID 2 ← in disuso

RAID 3

RAID 4

RAID 5 ← il piu popolare

RAID 6

RAID 7… ← proprietary RAIDs

Requisiti dei dischi

Numero *minimo* di dischi

- Tecnologie implementate
  - RAID 0 (striping): >= 2 dischi
  - RAID 1 (mirroring): (>)= 2 dischi
  - Striping+parity: > 3 dischi

Numero *massimo* di dischi

- Limitato dal controller

Il funzionamento ideale è con dischi:

- Identici
- Stessa capacità

Multiple level

Scelta livello RAID

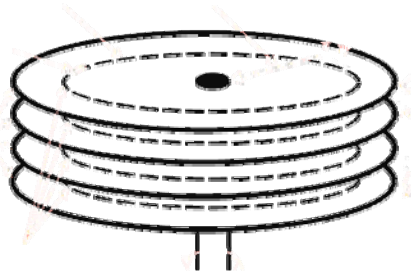
- Costi
- Prestazioni
- Affidabilità
- Complessità
- …

Differenti macchine

Array logici

Multiple level

Applicazioni/utenti con differenti requisiti → Difficile scelta del livello di RAID → Array separati



RAID 0: striping

Striping

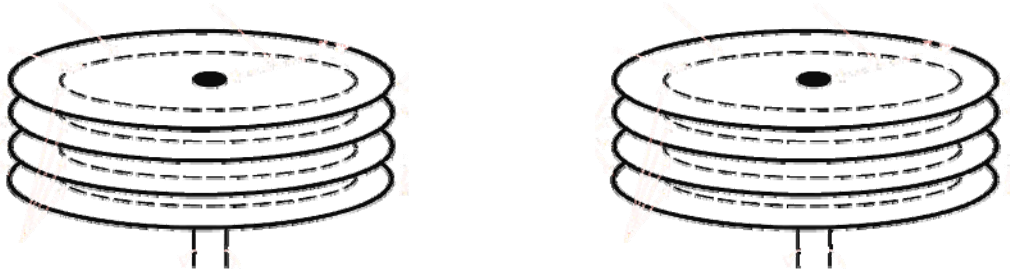
No ridondanza

~~Redundant~~ Arrays of Independent Disks

Dati

- Suddivisi in blocchi sequenziali
- Algoritmo di *striping* per distribuirli tra i dischi fisici

Numero minimo di dischi: 2



RAID 0: striping

Vantaggi

Costo minimo di implementazione

- Massima capacità
- No ridondanza

Elevate prestazioni

- Parallelismo di dischi e canali

Write efficienti

- Non c'è ridondanza da aggiornare

Svantaggi

No dischi *hot spares*

Bassa affidabilità

- No fault-tolerance
- No correzione errori

RAID 1: mirroring

Tutti i dati sono *duplicati* su un altro disco

- *Mirroring* (replica del disco)
- *Duplexing* (replica di disco e controller)

Numero minimo di drive: 2

Vantaggi

Elevata affidabilità

- Fault-tolerance

Read efficienti

- Tempo minimo tra i due drive
- Letture parallele (se un device è occupato si usa l'altro)

Svantaggi

Costo

Sfruttamento del 50% della capacità fisica

Write lente:

- Attesa del drive più lento

RAID 0+1

Tecnologie applicate ai dati:

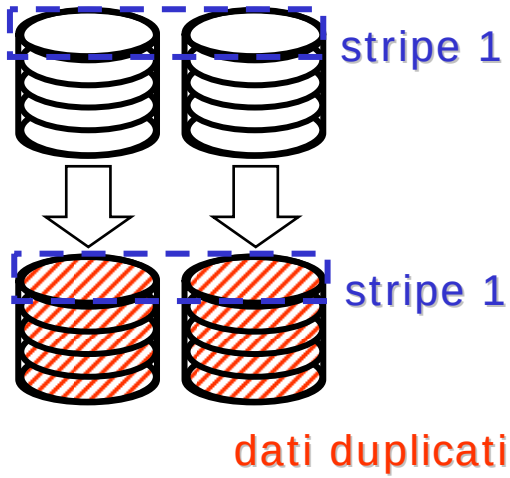
1) Striping
2) Mirroring

High data transfer performance

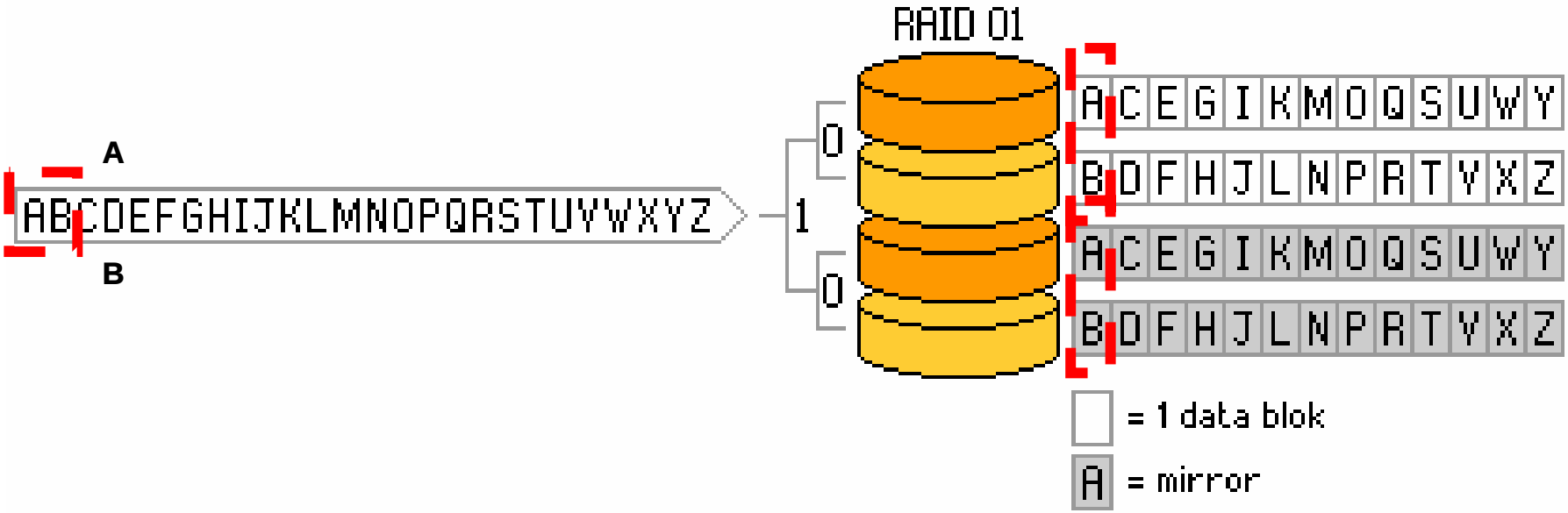
Buona affidabilità

- Il guasto di un disco porta alla situazione RAID 0

Overhead elevato



RAID 0+1


RAID 01
A
B
ABCDEFGHIJKLMNOPQRSTUVWXYZ
1
0
0
A C E G I K M O Q S U W Y
B D F H J L N P R T V X Z
A C E G I K M O Q S U W Y
B D F H J L N P R T V X Z
= 1 data blok
A = mirror

RAID 1+0

Tecnologie applicate ai dati:

1) Mirroring
2) Striping

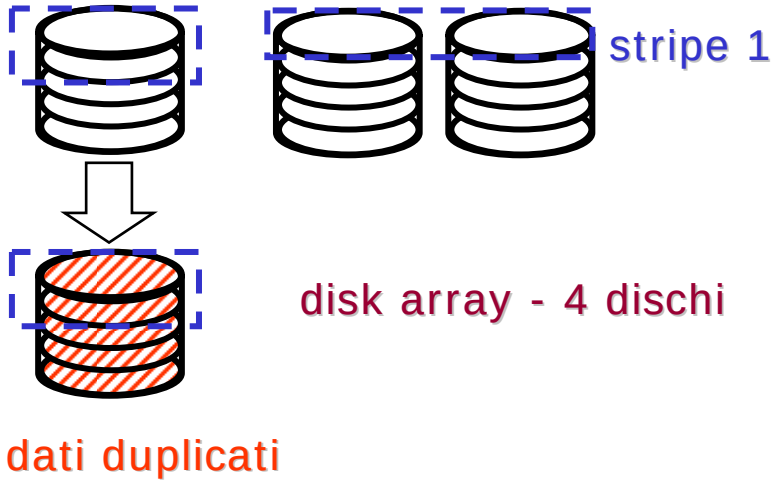
Elevate prestazioni

Fault-tolerance

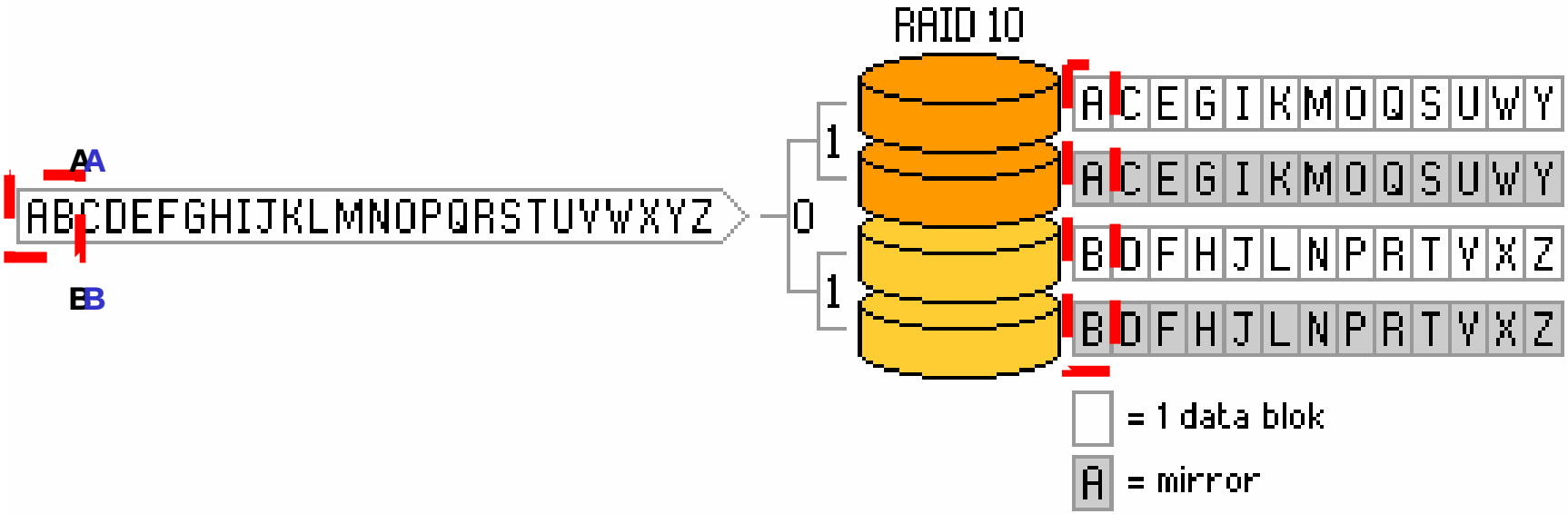
Numero minimo di dischi: 4

Costo

- Sfrutta solo 50% capacità fisica



RAID 1+0


RAID 10
AA
ABCDEFGHIJKLMNOPQRSTUVWXYZ
BB
0
1
1
A C E G I K M O Q S U W Y
A C E G I K M O Q S U W Y
B D F H J L N P R T V X Z
B D F H J L N P R T V X Z
= 1 data blok
A = mirror

RAID 0+1 vs 1+0

Fault-tolerance

Performance

Recovery

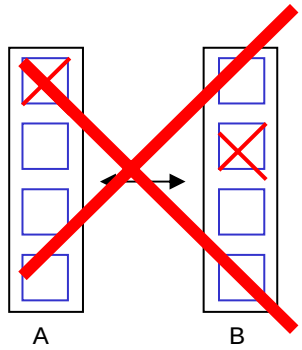
RAID 0+1 vs 1+0: Fault-Tolerance

Fault-Tolerance

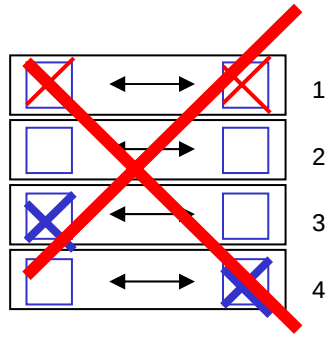
Performance

Recovery

RAID 0+1



RAID 1+0



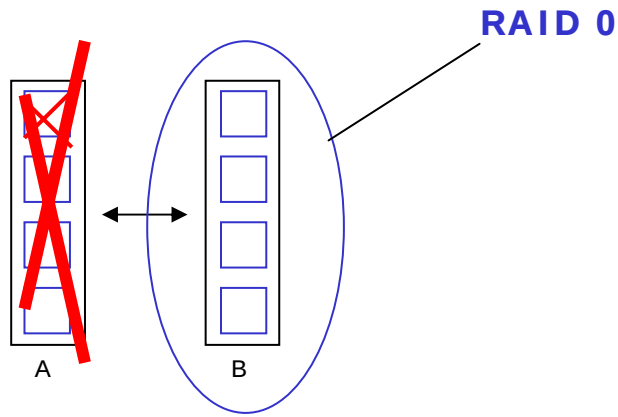
RAID 0+1 vs 1+0: Performance (a seguito di un guasto)

Fault-Tolerance

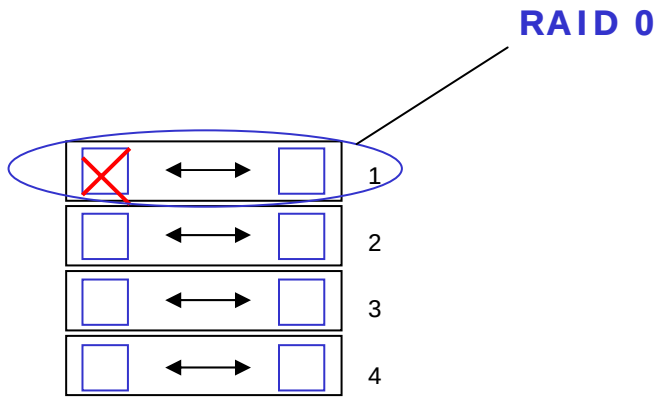
Performance

Recovery

RAID 0+1



RAID 1+0



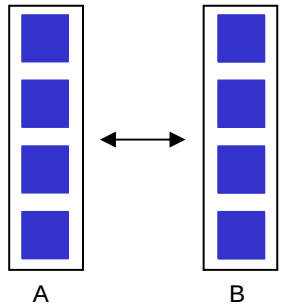
RAID 0+1 vs 1+0: Recovery

Fault-Tolerance

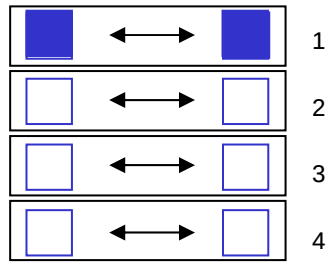
Performance

Recovery

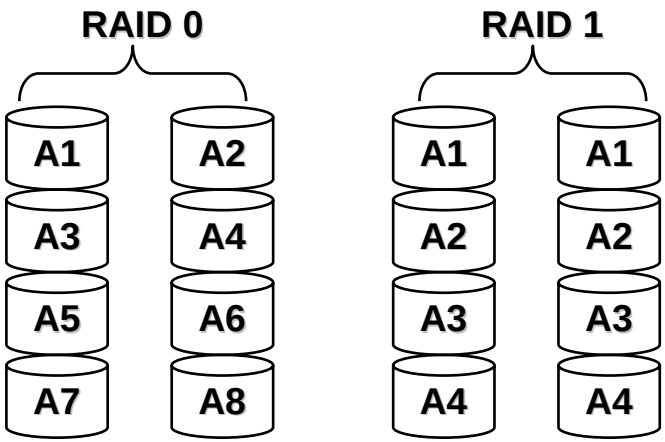
RAID 0+1



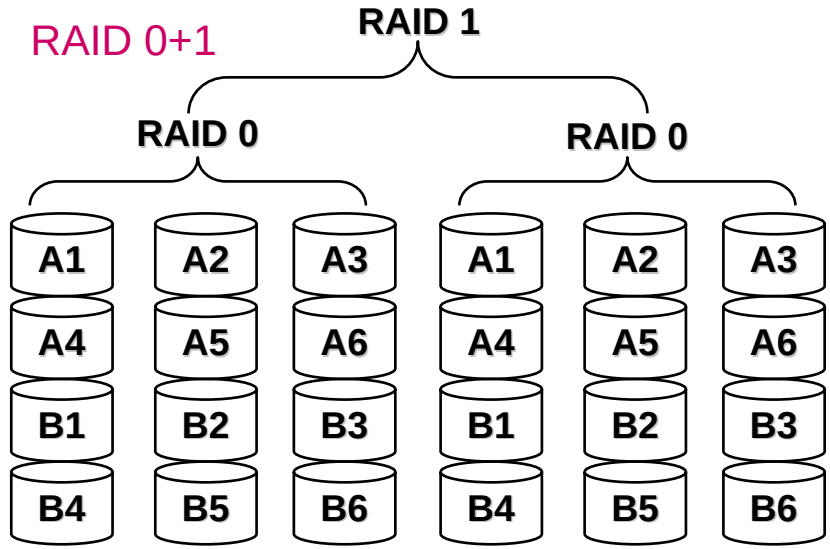
RAID 1+0



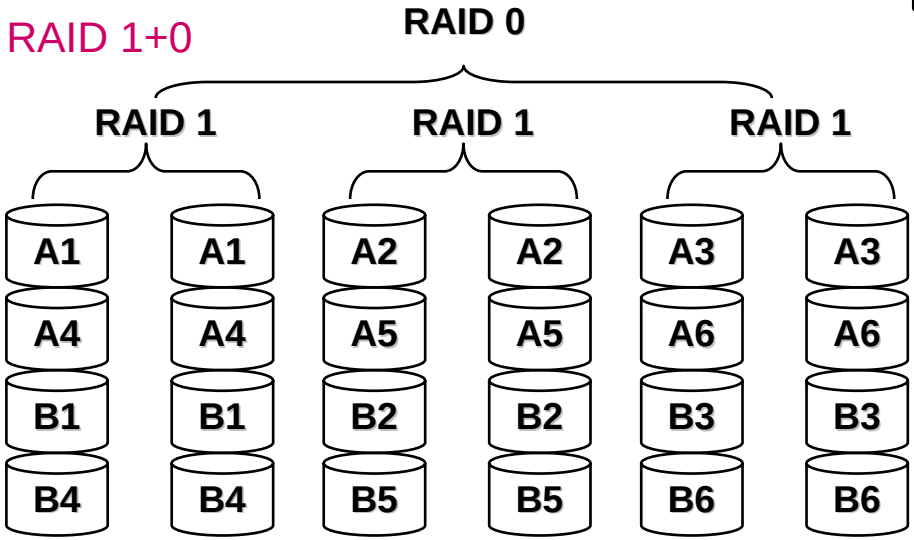
schema RAID 0, 1, 0+1, 1+0



RAID 0+1



RAID 1+0



A1, B1,... rappresenta un blocco di dati

Impianti Informatici

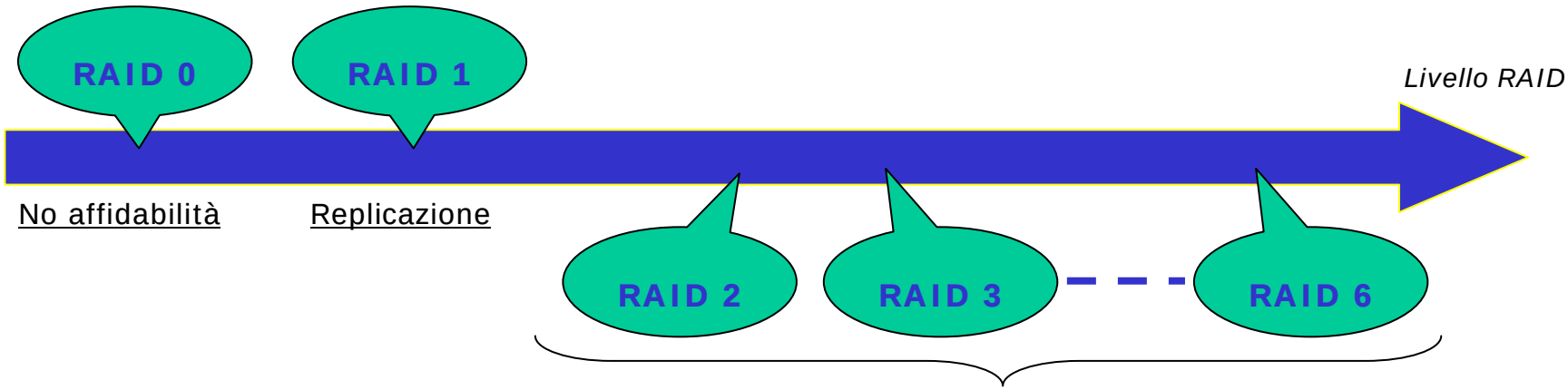
POLITECNICO DI MILANO





RAID

Livelli di RAID avanzati



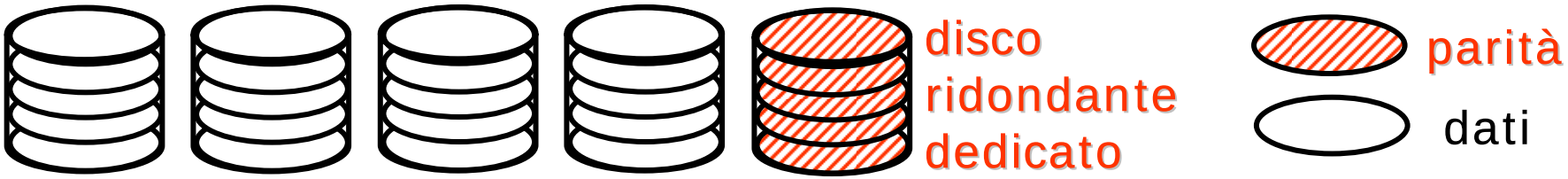
Combinano

- Fault-tolerance
- Error-correction
- Buone Prestazioni

RAID 4: block interleaved parity

Unità elementare: *blocco*

- read inferiori ad un blocco usano un solo disco

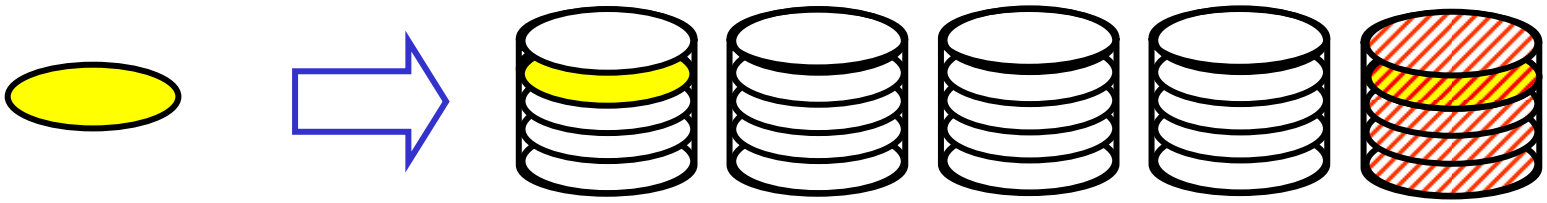


RAID 4: funzionamento

Operazione di *write*:

- Lettura dei dati nuovi (in input)
- Lettura dei dati vecchi (presenti su disco)
- Lettura della parità

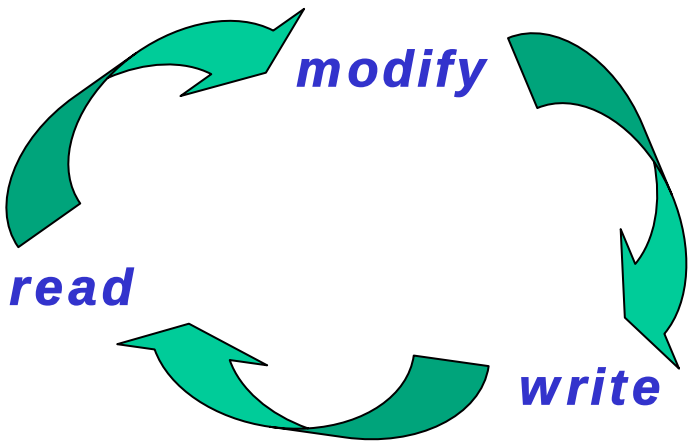
- Calcolo della nuova parità
- Aggiornamento blocchi



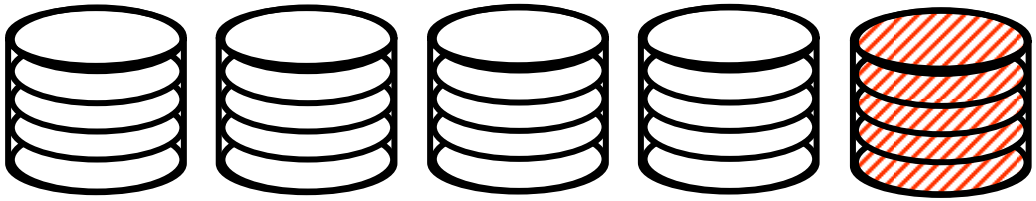
Ciclo *Read-modify-write*

Sistema **read-modify-write**

- per ogni operazione di scrittura breve
- 4 accessi
  - due per leggere i dati vecchi e la parità vecchia
  - uno per scrivere i dati nuovi
  - uno per scrivere la parità nuova



RAID 4: caratteristiche



Disco ridondante

- Acceduto da ogni write
- Possibile bottleneck

Affidabilità

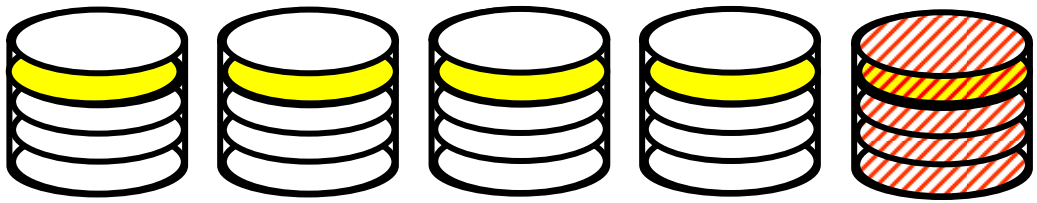
- RAID 4 guasto con due dischi guasti

È possibile usare dischi *hot spares*

RAID 4: prestazioni

Lettura

- Veloce
- Parallelismo (raramente è richiesto l'accesso a disco di ridondanza x checkout dati in fase di lettura)

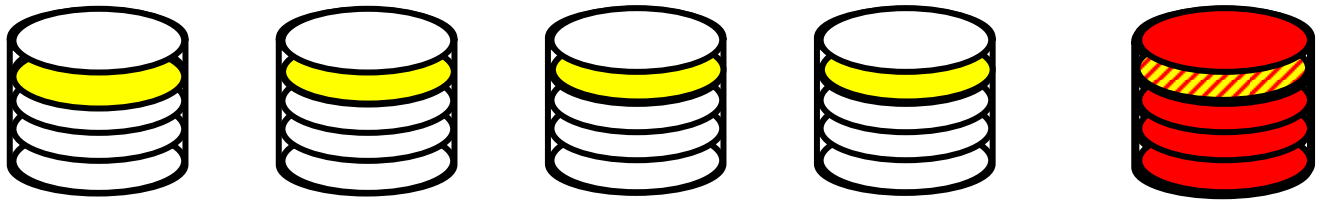


Scrittura

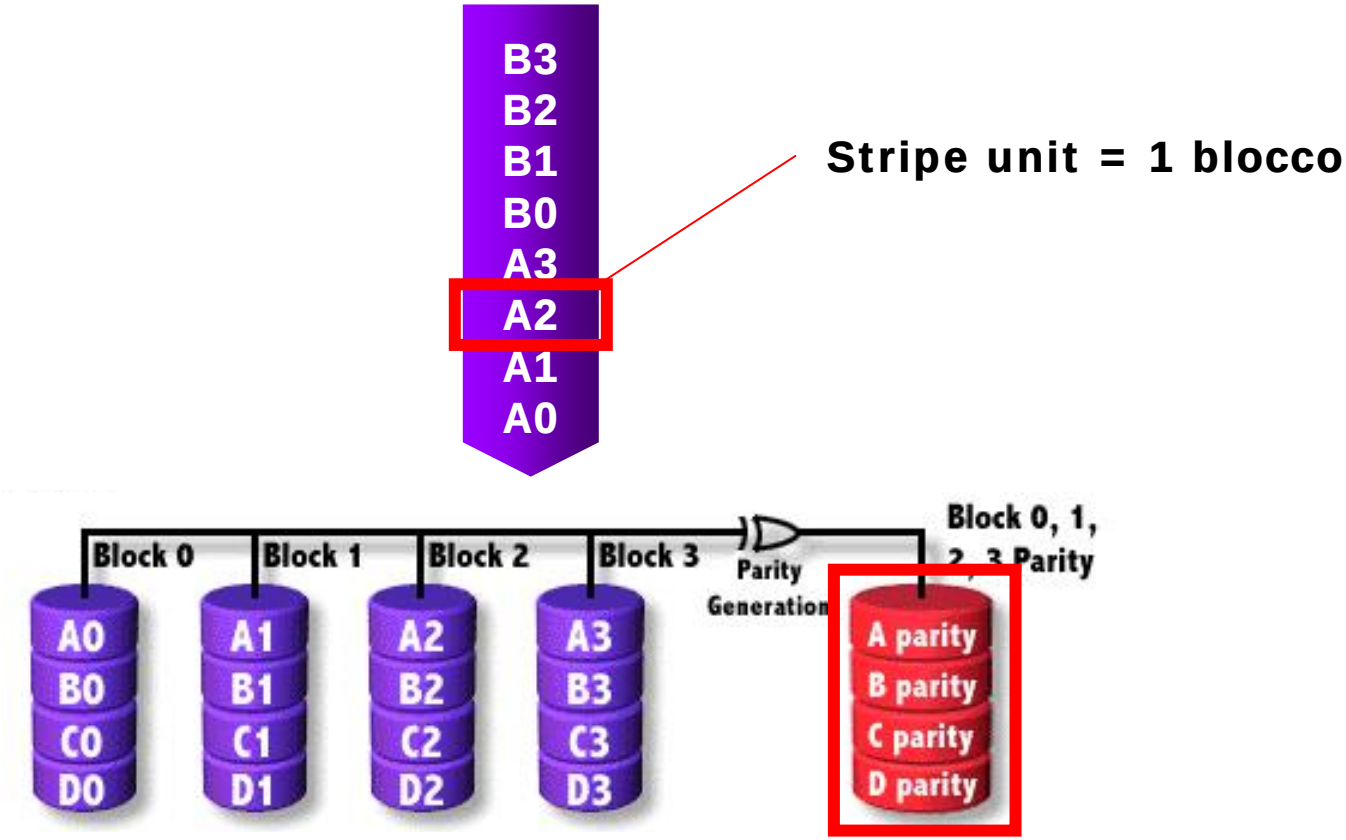
- Lenta
- Penalizzata dal *parity block*

RAID 4: distribuzione dei dati

	Disco 1	Disco 2	Disco 3	Disco 4	Disco 5 ridondante
Stripe 1	*Block 1*	*Block 2*	*Block 3*	*Block 4*	Parity 1-4
Stripe 2	*Block 5*	*Block 6*	*Block 7*	*Block 8*	Parity 5-8
Stripe 3	*Block 9*	*Block 10*	*Block 11*	*Block 12*	Parity 9-12



RAID 4: distribuzione dei dati



RAID 5: block interleaved distributed parity

Soluzione ampiamente adottata

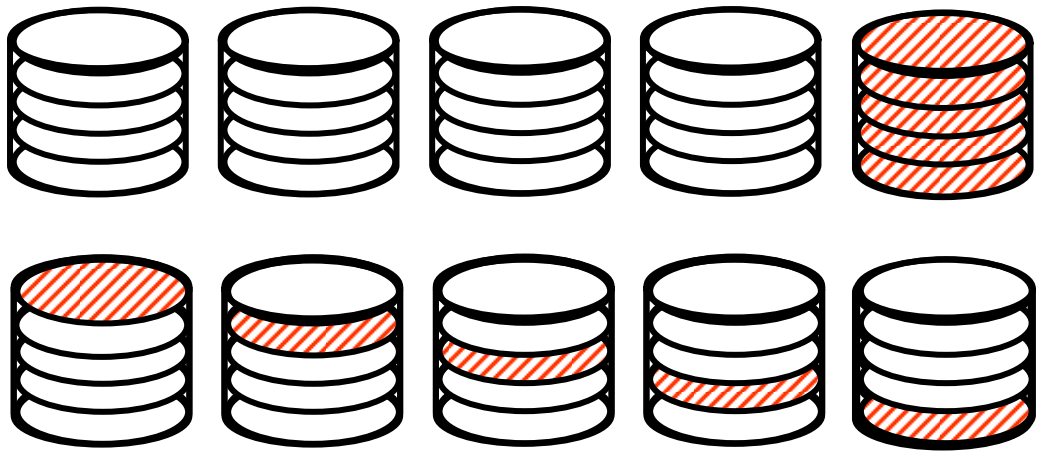
Versatile

- Prestazioni/affidabilità
- Costo minimo per la ridondanza

Blocchi di parità distribuiti su tutti i dischi fisici

RAID 4

RAID 5



RAID 5: prestazioni

Write

- Più lente di RAID 0 e RAID 1
- Occorre scrivere su tutti i dischi

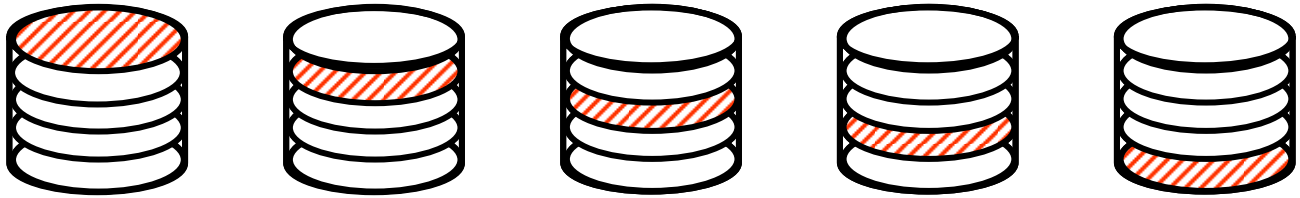
Read

- Più veloci di RAID 1
- Parallelismo

- In genere i blocchi di parità non sono acceduti nell'operazione di lettura.
- Vengono letti se un settore dà luogo a un errore *CRC* (Cyclic Redundancy Check): il settore errato viene ricostruito utilizzando le informazioni dei rimanenti blocchi della stripe unit in questione e del blocco di parità

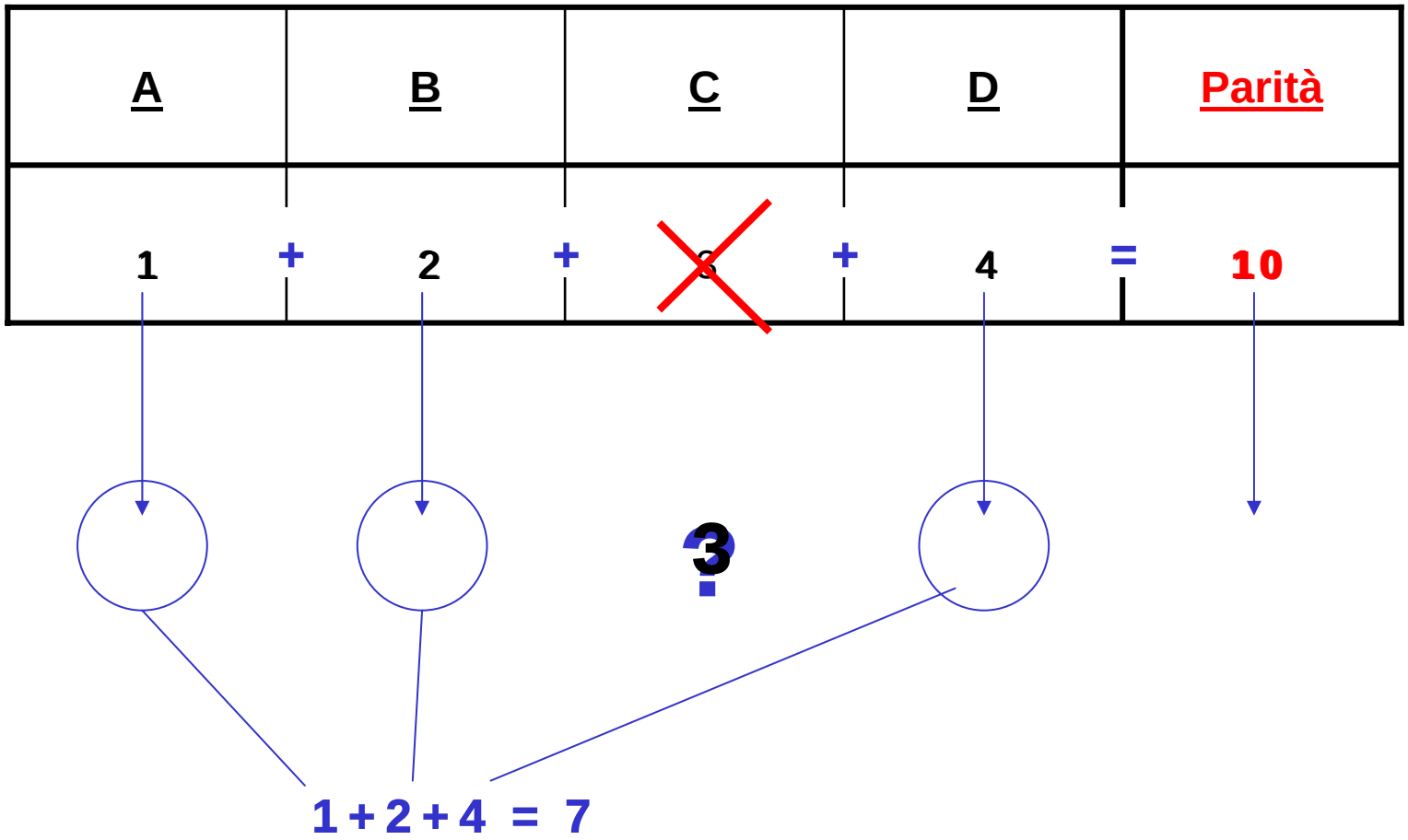
Load balancing su tutti i dischi

RAID 5: scrittura



	Disco 1	Disco 2	Disco 3	Disco 4	Disco 5
Stripe 1	*Block 1*	*Block 2*	*Block 3*	*Block 4*	**Parity 1-4**
Stripe 2	*Block 5*	*Block 6*	*Block 7*	**Parity 5-8**	*Block 8*
Stripe 3	*Block 9*	*Block 10*	**Parity 9-12**	*Block 11*	*Block 12*

Calcolo della parità



Esempio di ridondanza

Disco 1	Disco 2	Disco 3	dati ridondanti
10	8	2	**20**
10	guasto	2	**20**

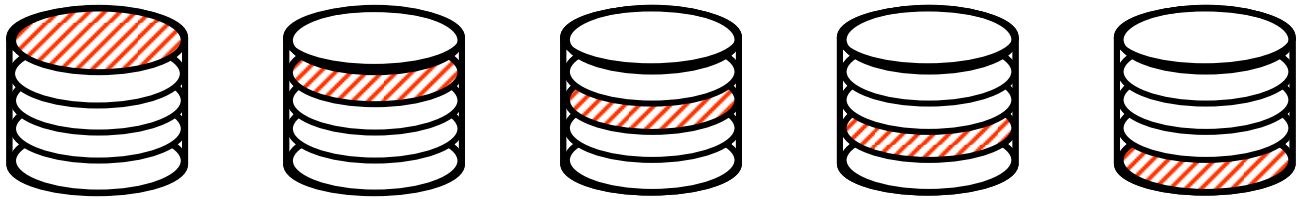
guasto = 20 – (10 + 2) = 8

Disco 1	Disco 2	Disco 3	parità
1	1	0	**0**
1	guasto	0	**1**

parità = somma modulo 2

guasto = parità – (1+0) = 0

RAID 5: scrittura (esempio)



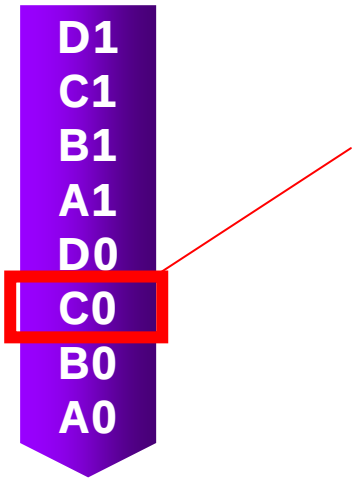
	Disco 1	Disco 2	Disco 3	Disco 4	Disco 5
Stripe 1	*Block 1*	*Block 2*	*Block 3*	*Block 4*	**Parity 1-4**
Vecchio	110	011	111	100	110
Nuovo	*011*	*011*	*111*	*100*	

011

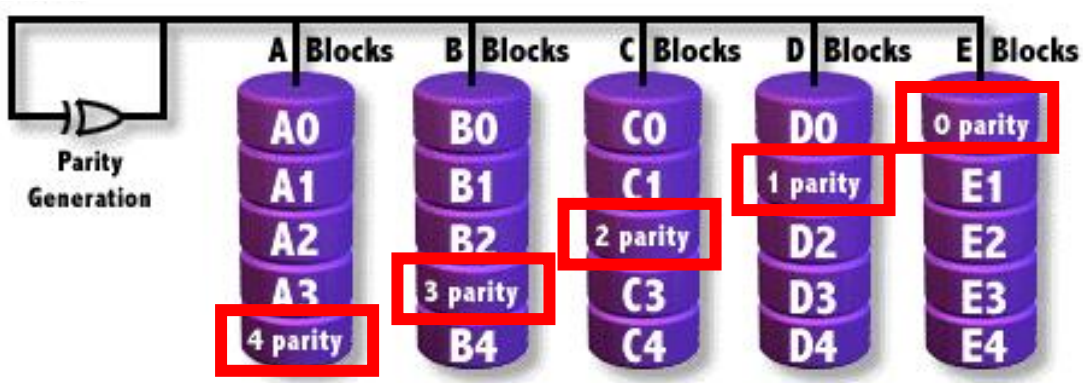
parità

dati

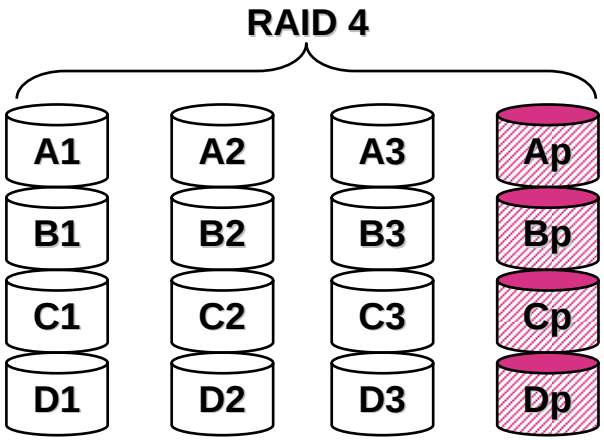
RAID 5: distribuzione dei dati

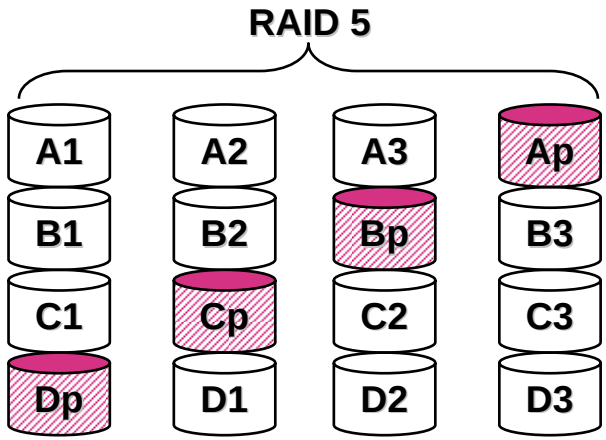


Stripe unit = 1 blocco

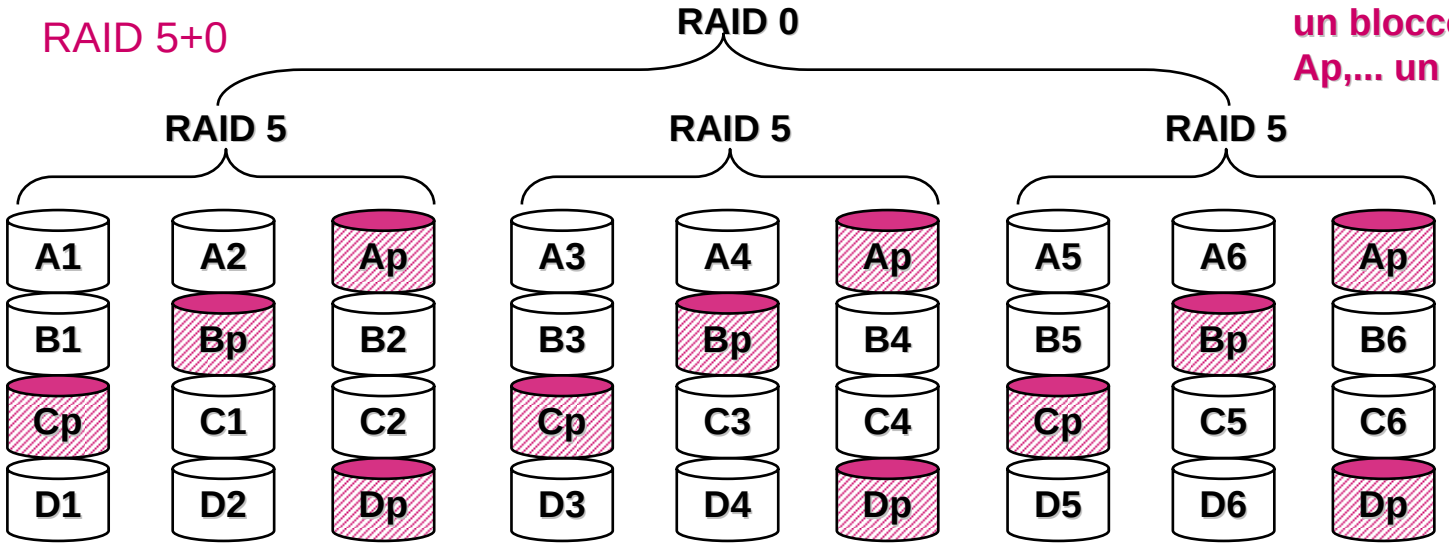


Schema RAID 4, 5, 5+0





RAID 5+0



A1, B1,... rappresenta un blocco di dati
Ap,... un blocco di parità

Impianti Informatici

POLITECNICO DI MILANO





RAID

Livelli meno diffusi

RAID 2

RAID 3

RAID 6

RAID 2

RAID 2

RAID 3

RAID 6

Striping

- Dati divisi a livello di *bit*

Ridondanza

- Codici di *Hamming*
- In lettura viene verificata la correttezza dei dati e corretti gli errori su un singolo drive
- Individua 2-bit errors e corregge 1-bit errors on the fly

Affidabilità

Capienza

- con 8 dischi è 5 volte più capiente

Velocità (teorica)

- con 8 dischi è 5 volte più veloce
- non per accessi “piccoli”

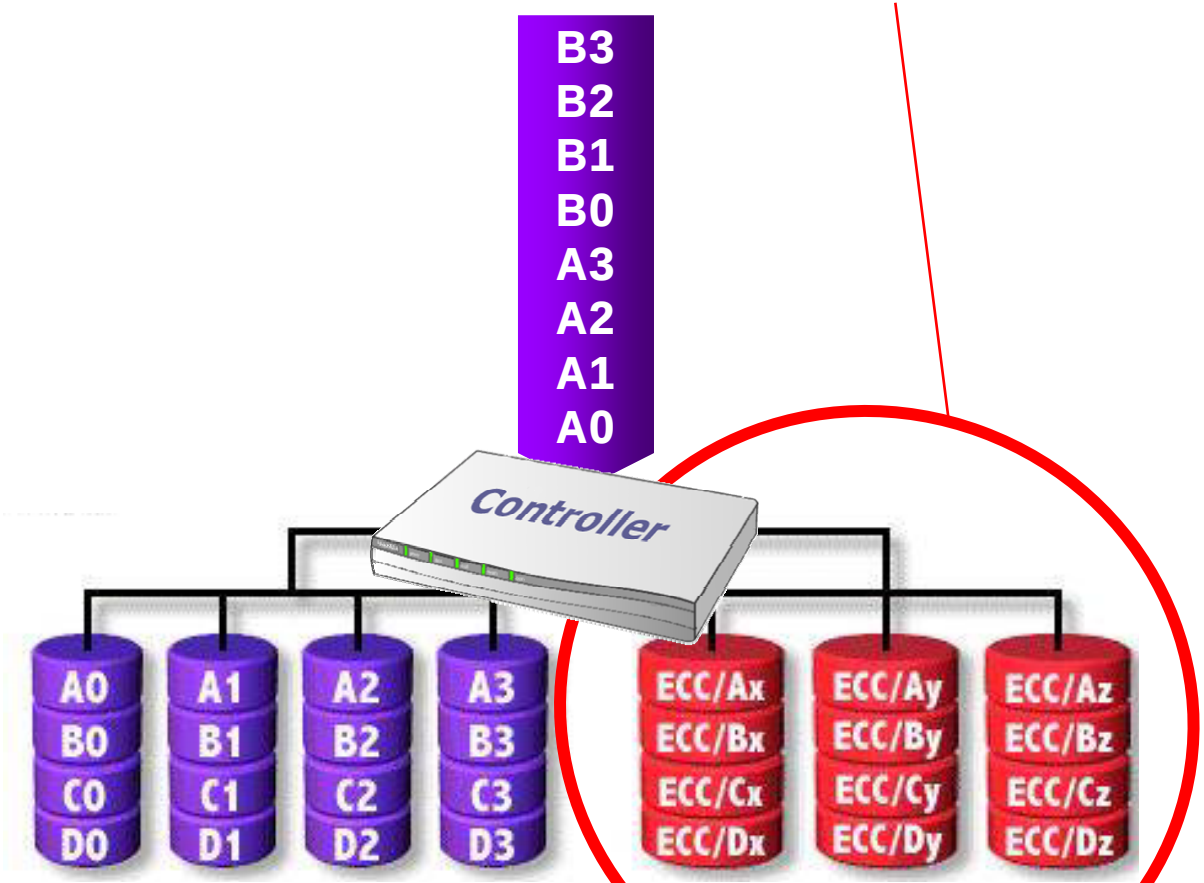
RAID 2

#dischi ridondati = $\log_2$ #dischi dati

RAID 2

RAID 3

RAID 6



RAID 3

RAID 2

RAID 3

RAID 6

Striping

- Dati divisi a livello di *byte*

Ridondanza

- Informazioni di parità

Affidabilità

Capienza

- con 5 dischi è 4 volte più capiente

Velocità (teorica)

- con 5 dischi è 4 volte più veloce
- non per accessi “piccoli”

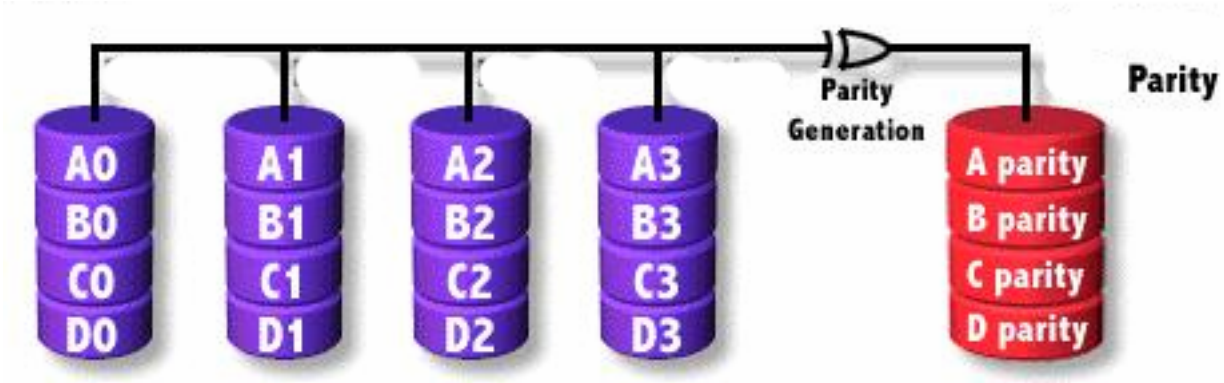
RAID 3

RAID 2

RAID 3

RAID 6

- Un solo disco per i bit di parità
- Adatto per applicazioni che richiedono elevata banda ma medio I/O rate
- Una sola richiesta di I/O viene eseguita per volta
- Ogni read accede a tutti i dischi dati
- Ogni write accede a tutti i dischi dati ed al disco di parità



RAID 3

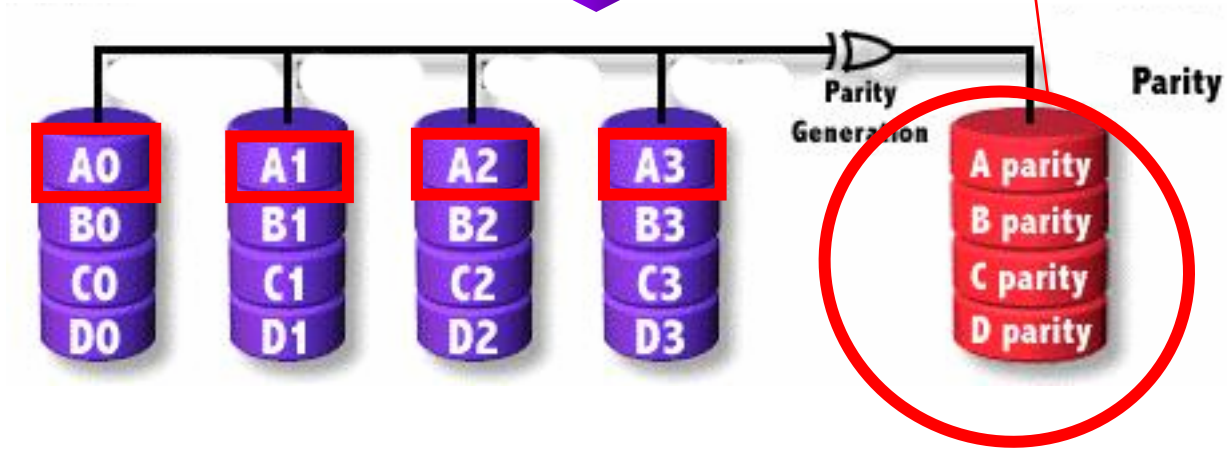
RAID 2

RAID 3

RAID 6

B3
B2
B1
B0
A3
A2
A1
A0

Disco di parità



RAID 6

RAID 2

RAID 3

RAID 6

Striping

- Dati divisi a livello di *blocco*

Ridondanza

- Informazioni di parità
- Distribuita su tutti i dischi

Doppia ridondanza

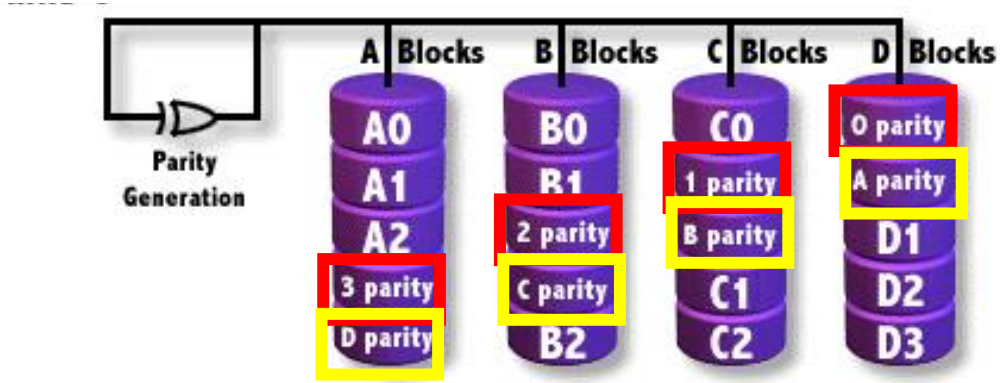
- Due parità indipendenti
- Alta affidabilità

RAID 6: P+Q

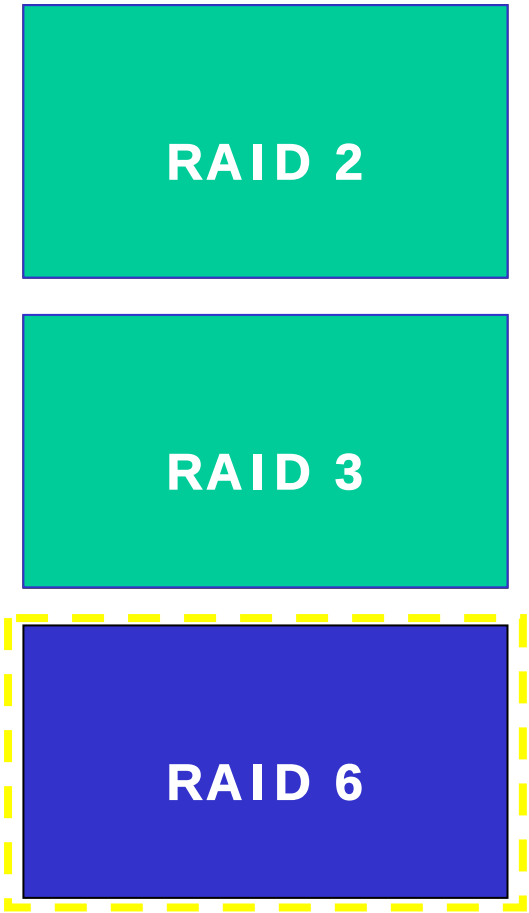
RAID 2

RAID 3

RAID 6



RAID 6: read-modify-write



Operazione di scrittura “breve”

↓

Sistema *read-modify-write*

↓

6 accessi al disco

Ridondanza P

Ridondanza Q

RAID 6: calcolo di P+Q

RAID 2

RAID 3

RAID 6

P

1	+	2	+	3	+	4	=	*10*
5		6		7		8		*26*
9		10		11		12		*42*
15		*18*		*21*		*22*		

← Q

Il RAID 6 tollera il guasto di 2 dischi

Impianti Informatici

POLITECNICO DI MILANO





RAID

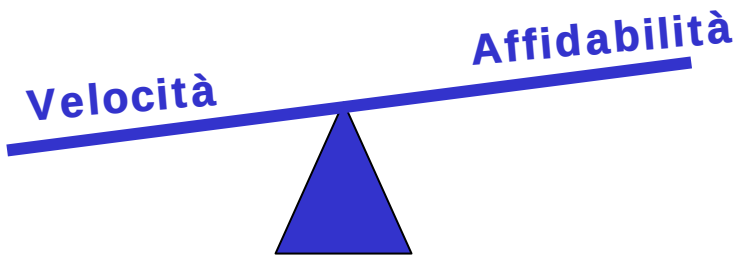
Criteri di scelta

Velocità

- I/O di scrittura
- I/O di lettura

Affidabilità

- Fault-tolerance
- Error correction



Criteri di scelta del livello di RAID

Velocità

- I/O di scrittura
- I/O di lettura
- tempi di recovery

Parallelismo

Affidabilità

- Fault-tolerance
- correzione errori

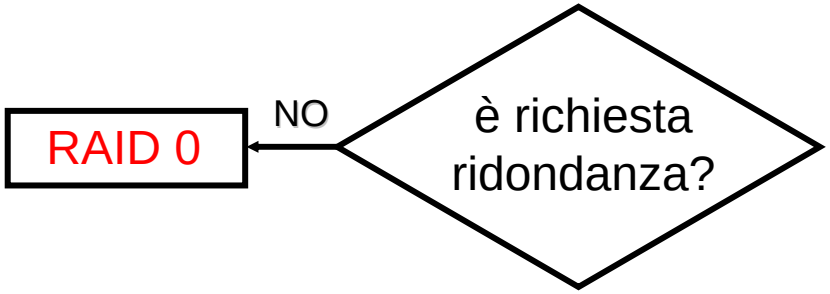
Ridondanza

Duplicazione

Costi

- sfruttamento della capacità fisica
- tipi di soluzione
- caratteristiche controller

Ridondanza



Bassa affidabilità

Read/write veloci

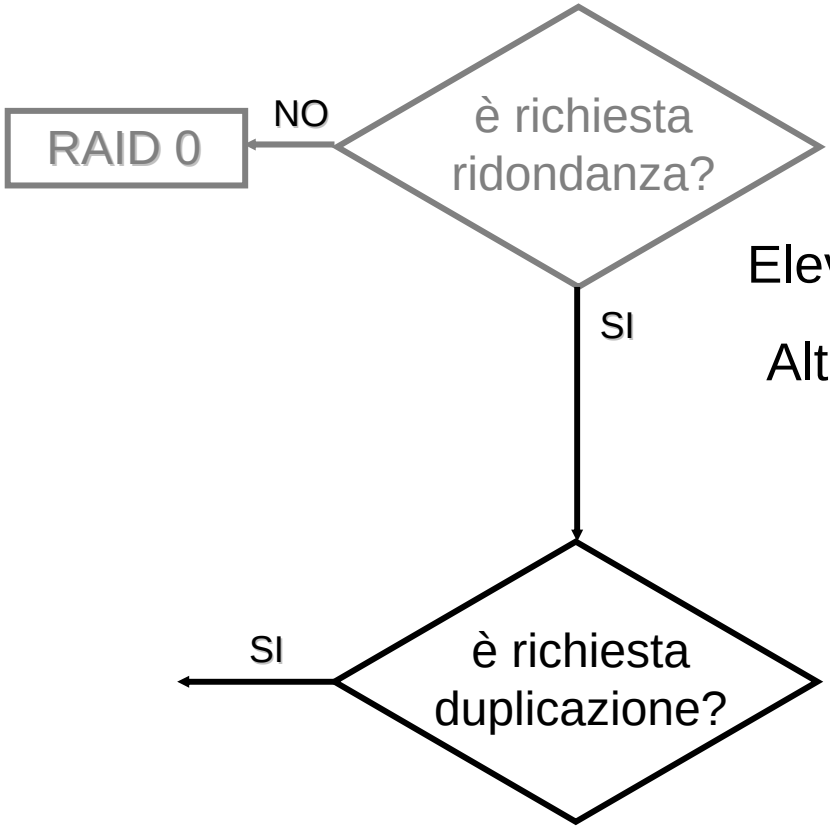
- Parallelismo

Sfruttamento capacità fisica

- Basso costo

HPC (High-Performance Computing)

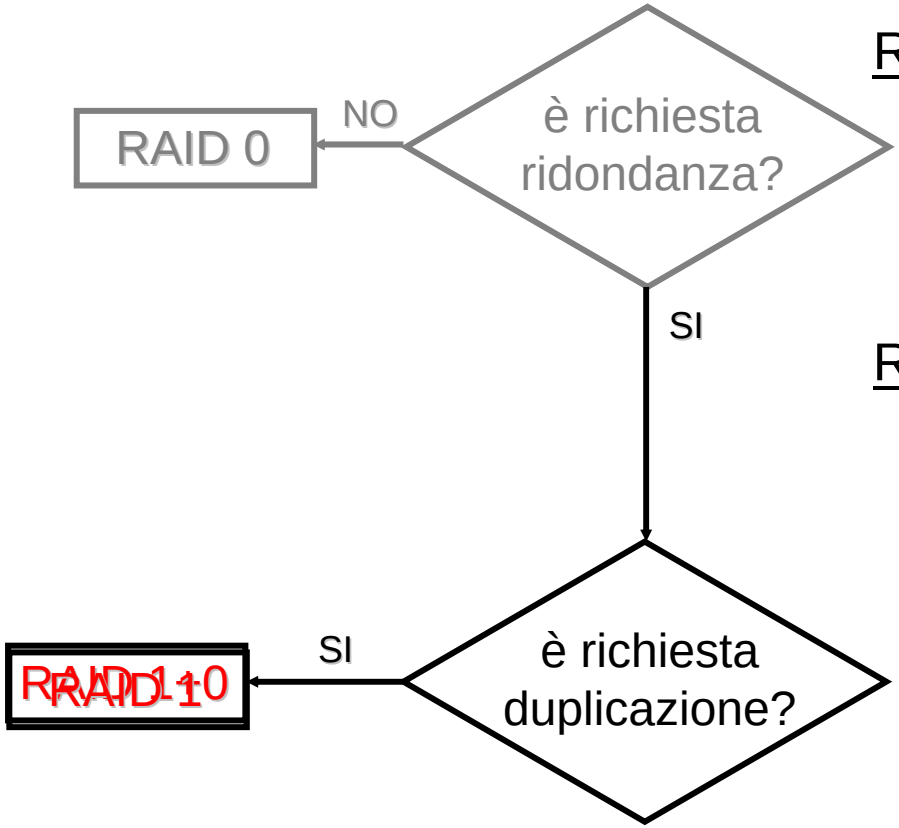
Duplicazione



Elevata affidabilità

Alto costo

Duplicazione: RAID 1+0



RAID 1 (*mirroring*):

- Massimo due dischi

RAID 1+0:

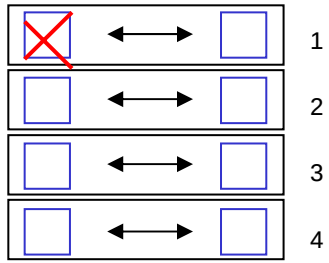
- Mirroring+ Striping
- Migliore di RAID 0+1

- Velocità
  - Letture brevi

Database

RAID 1+0

RAID 1+0



Affidabilità

- Fault-tolerant con 1 disco rotto
- Tollerante anche a rotture di più dischi, purché di differenti mirror

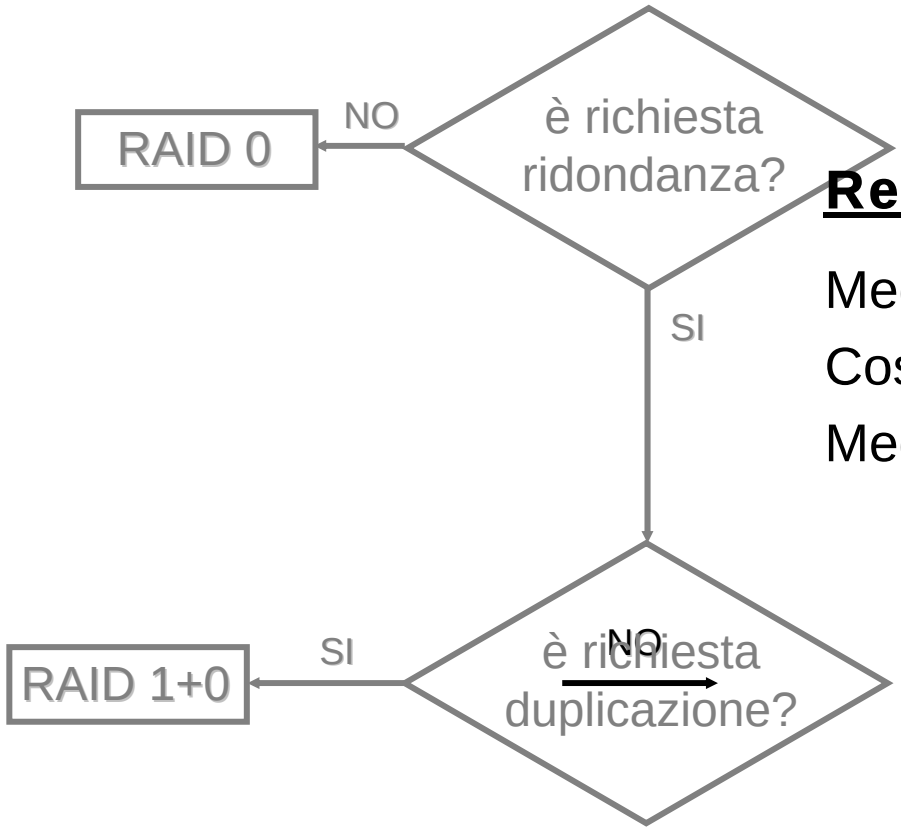
Prestazioni

- Meno efficiente di RAID 1
- *Mirroring + Striping*

Confronto livelli 0+1, 1+0

- La disposizione dei blocchi è identica se non per i dischi che sono in un diverso ordine
- Alcuni controller 0+1 combinano in un'unica operazione striping e mirroring
- **0+1**
  - non tollera due guasti simultanei (eccetto nel caso in cui interessino la stessa stripe)
  - nel caso di guasto a un singolo disco, qualunque guasto ad altra stripe è un *single point of failure*
  - il ripristino del disco richiede la partecipazione di tutti i dischi dell'array
- **1+0**
  - un disco per ogni gruppo RAID 1 può guastarsi ma se non riparato, l'altro disco è *single point of failure* dell'intero array

Duplicazione: contro



Requisiti:

Media affidabilità

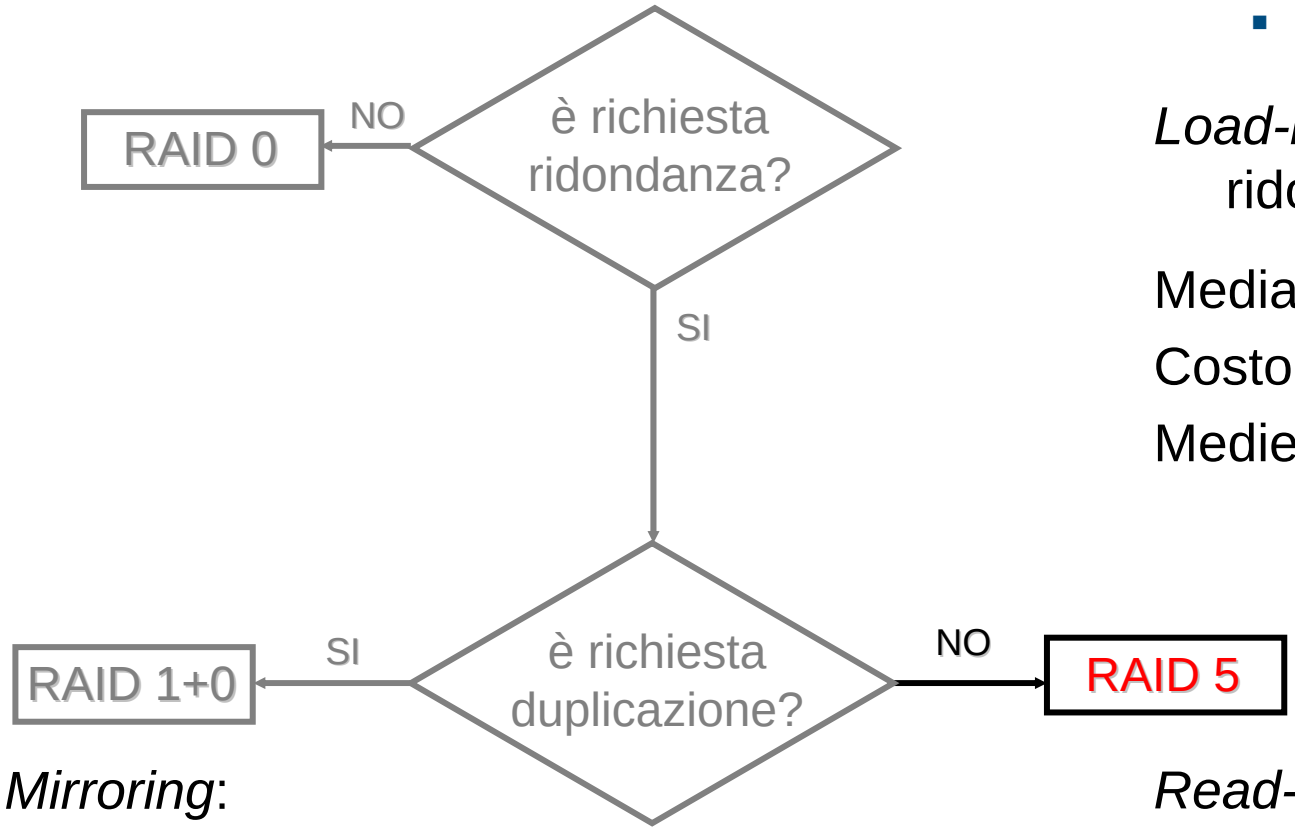
Costo medio/basso

Medie prestazioni

Mirroring:

- Usa 50% capacità fisica

RAID 5



Striping

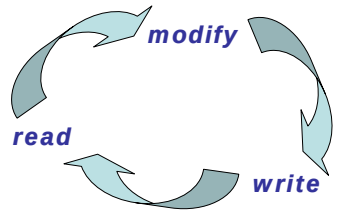
- *Read* parallele

Load-balancing della ridondanza

Media affidabilità

Costo medio/basso

Medie prestazioni



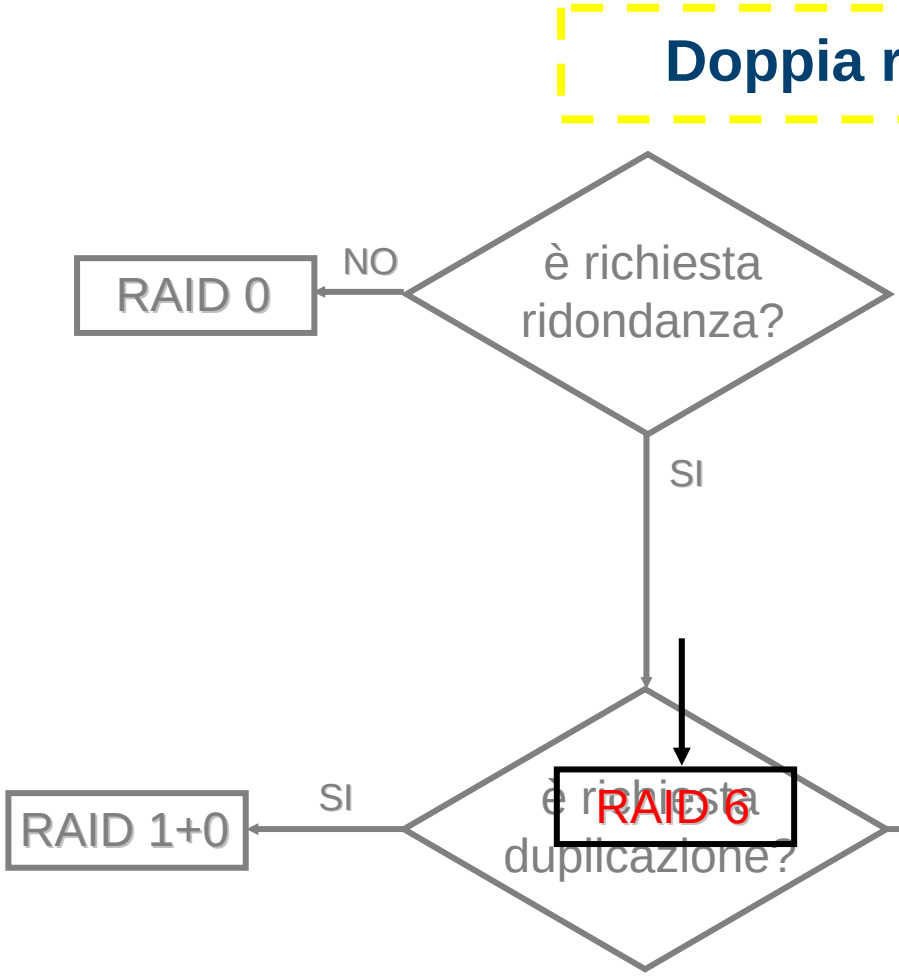
Mirroring:

- Usa 50% capacità fisica

Read-modify-write

- Scritture brevi

Doppia ridondanza



Fault-tolerant al guasto di **2** dischi (per applicazioni molto critiche)

Prestazioni *write*

- Scritture brevi lente
- Update ridondanza
- 6 accessi al disco

Load-balancing della ridondanza