

Sistemi Operativi

Da Unix a GNU/Linux (parte 3)

Docente: Claudio E. Palazzi
cpalazzi@math.unipd.it

Crediti per queste slides al Prof. Tullio Vardanega

Caratteristiche del *File System* – 1

- *File* visto da FS come **sequenza di *byte*** di significato arbitrario
 - Fissato dal programma applicativo
- *File* regolari, *file* repertorio (*directory*) e *file* speciali che mappano dispositivi di I/O
- Nome inizialmente limitato a 14 caratteri (UNIX v7)
- Poi esteso fino a 255 (UNIX BSD → GNU/Linux)
 - Estensione **non** obbligatoria
 - Convenzione di estensione a scelta del programma applicativo e/o dell'utente

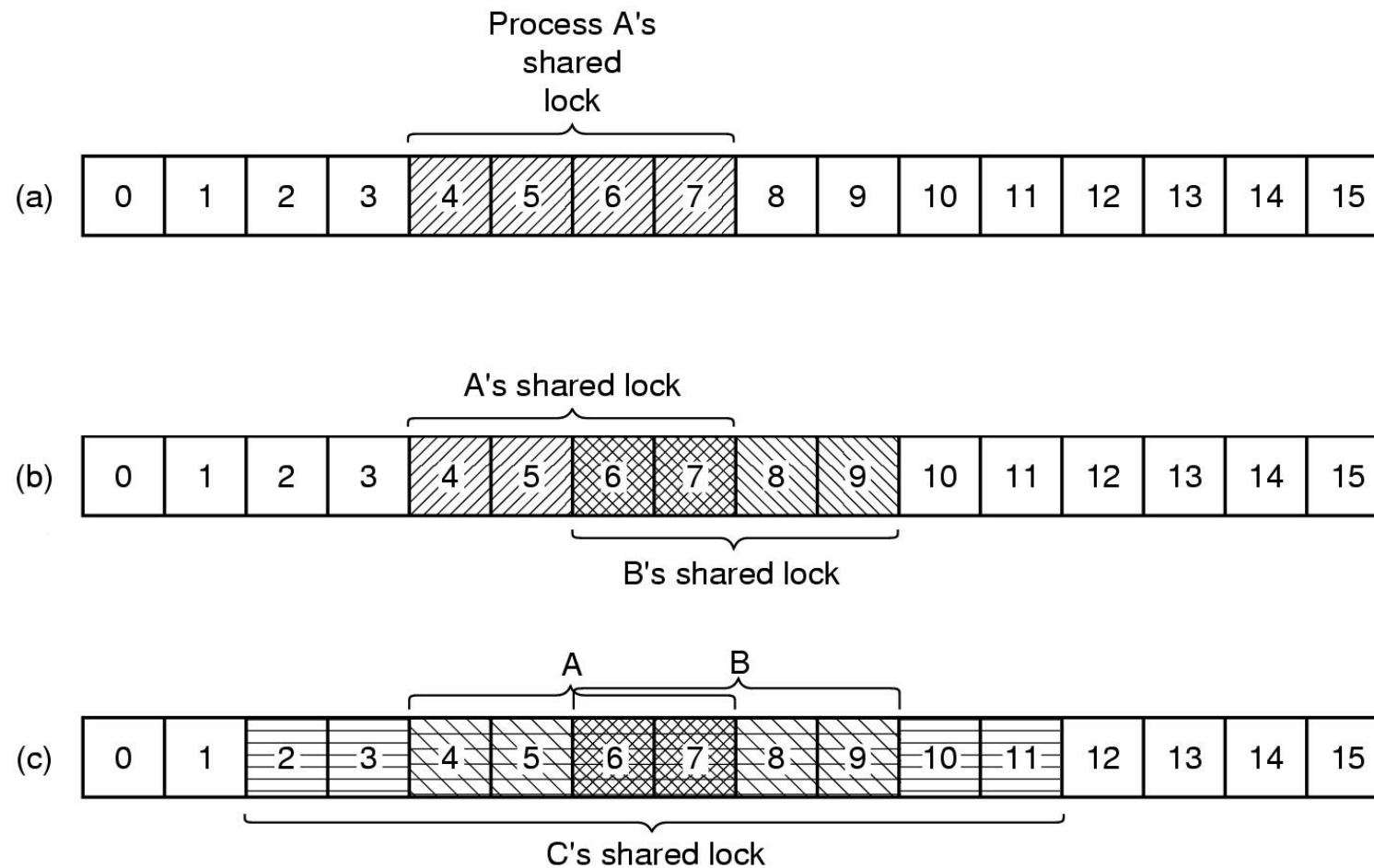
Caratteristiche del *File System* – 2

- *File* designato mediante cammino (*path*) assoluto o relativo
 - Il cammino relativo richiede la nozione di *directory* di lavoro corrente
 - *pwd* per visualizzarne la posizione assoluta
 - *cd* per cambiare posizione
 - Un intero FS **B** posto su una partizione visibile può essere ritenuto come parte di un FS **A** mediante *mount*
 - La radice di **B** viene designata con un nome (cammino) specifico in **A** detto *mount point*

Caratteristiche del *File System* – 3

- **Controllo di accessi concorrenti** (*locking*) - POSIX
 - A grana grossa (per *directory* o per *file*)
 - Mediante uso esplicito di semafori convenzionali
 - A grana fine (per gruppi di *byte* in un *file*)
 - Mediante meccanismi dedicati
- Due distinte modalità d'uso
 - **Accesso simultaneo condiviso** (*shared lock*)
 - Più accessi R alla stessa zona ma anche a zone solo parzialmente sovrapposte
 - **Accesso esclusivo** (*exclusive lock*)
 - Consente un solo accesso per zona selezionata

Caratteristiche del *File System* – 4

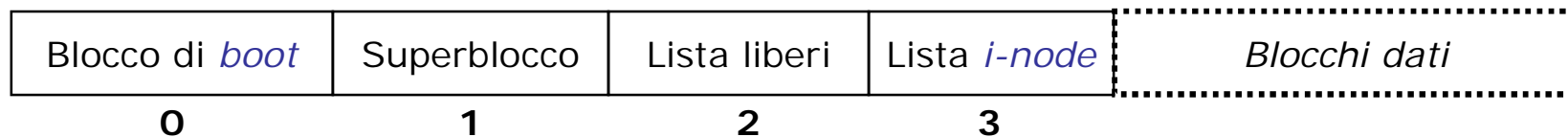


Esempi di chiamate di *File System*

- Disponibili all'utente solo **indirettamente** tramite incapsulazione in procedure di libreria
 - **lseek**
 - Fissa l'indice di posizione all'interno di un *file*
 - Come *offset* espresso in *byte* rispetto ad un riferimento dato
 - **Accesso diretto**
 - **stat**
 - Fornisce informazioni su *file* prelevandole dall'*i-node* corrispondente
 - Chiamata incapsulata dal comando **stat** di *shell*
 - Provare per esercizio dopo aver letto “**man stat**” 😊

Realizzazione del FS in UNIX – 1

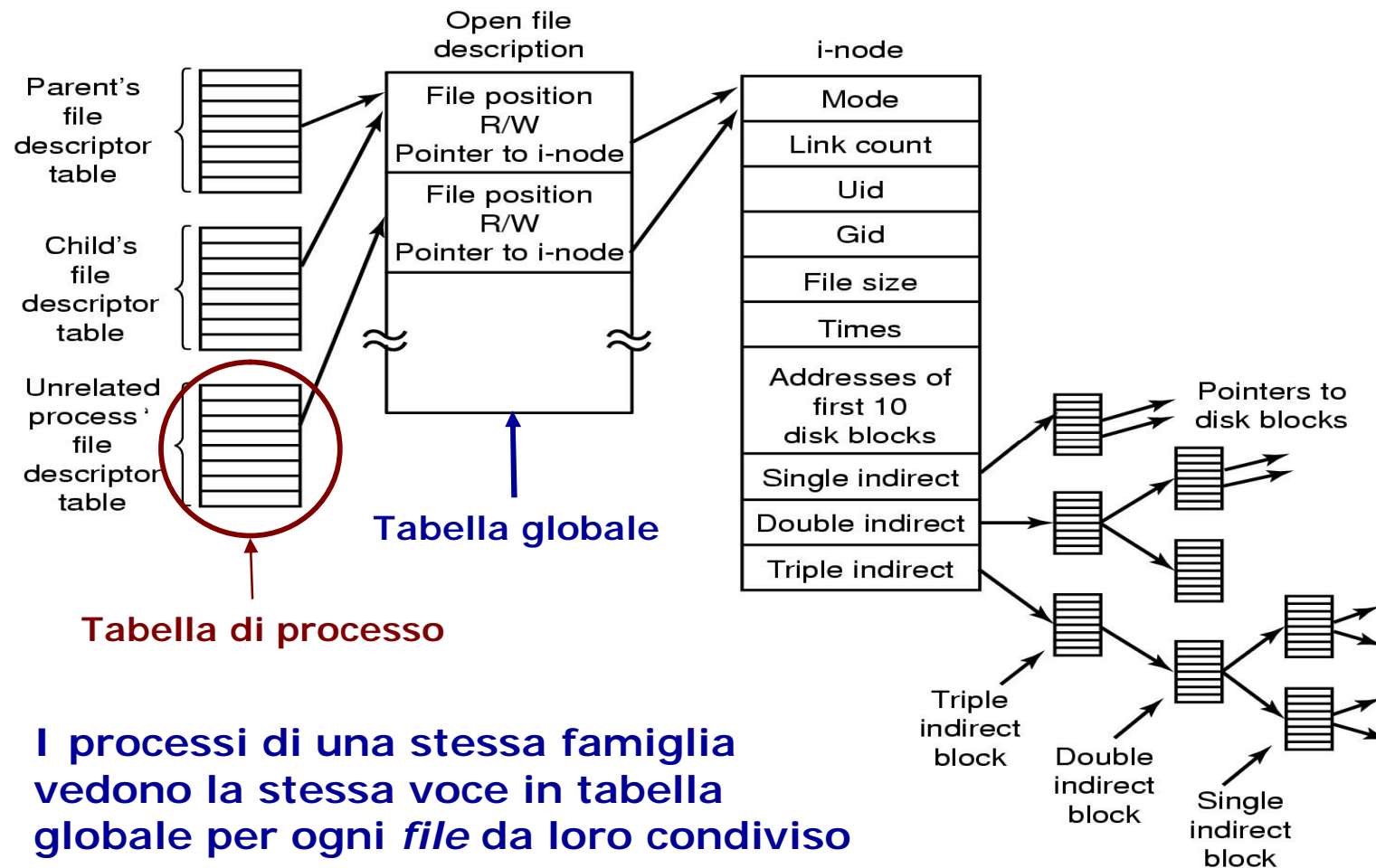
- **Struttura di partizione secondo UNIX v7**
- Il super-blocco (1) indica tra l'altro il # di *i-node* e di blocchi nel FS e fornisce il puntatore alla lista dei blocchi liberi (2)
- Gli *i-node* (3) sono numerati 1..N
- *Directory* come insieme **variabile** e **non ordinato** di unità informative (*entry*)
 - Ampie 16 B
 - 14 B (codifica ASCII) per nome di *file*
 - 2 B per numero di *i-node*



Realizzazione del FS in UNIX – 2

- In nucleo usa **due** strutture di controllo
 - Un insieme di **tabelle di processo** contiene “descrittori utente” dei *file* attualmente in uso a ciascun processo
 - A ogni descrittore utente deve corrispondere l'attuale posizione di R/W
 - Però ogni processo deve avere il suo proprio indice di posizione sui propri *file* aperti
 - Possono esistere **più** posizioni di R/W su uno stesso *file* condiviso
 - L'indice **non può** essere ritenuto nell'*i-node* che è **unico** per *file* !
 - Una **tabella globale** mantiene la corrispondenza tra tutti i *file* aperti e i loro *i-node*
 - Ciascuna voce nella **tabella di processo** punta a una voce nella **tabella globale** che specifica diritti e posizione di R/W corrente nel *file*
 - La stessa voce \forall *file* condiviso da processi di una stessa famiglia
 - Voce diversa per stesso *file* per processi non apparentati

Realizzazione del FS in UNIX – 3



Realizzazione del FS in UNIX – 4

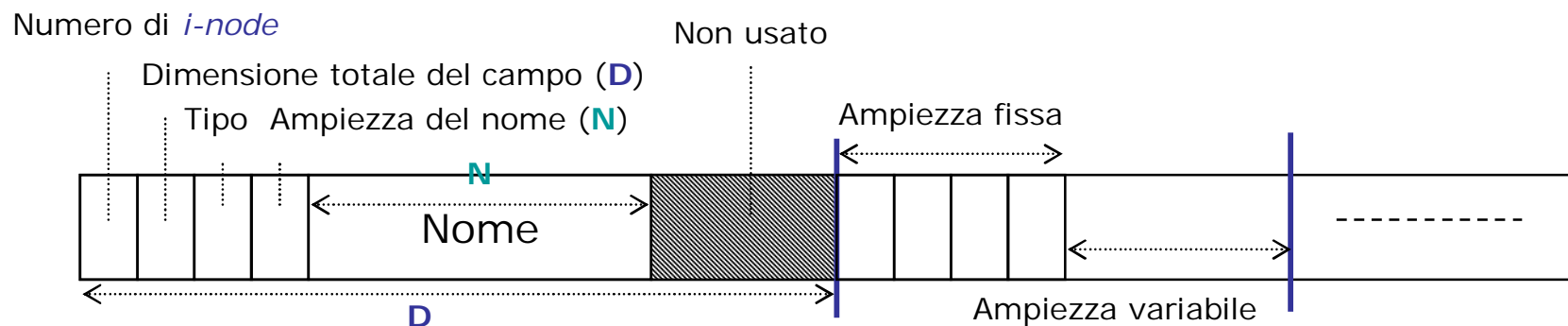
- L'*i-node* principale del *file* contiene (tra l'altro) l'indirizzo dei suoi primi 12 blocchi dati
 - 1 *i-node* ha la dimensione di 1 frazione di blocco (64 B)
- Per *file* più grandi 1 campo dell'*i-node* principale punta a 1 *i-node* secondario che contiene puntatori ad altri blocchi dati
 - *i-node* principale con campo *single-indirect*
- Per *file* ancora più grandi l'*i-node* secondario contiene puntatori a nodi *single-indirect*
 - *i-node* principale con campo *double-indirect*
- È previsto anche un campo *triple-indirect*

Esempio d'uso di *i-node* (UNIX v7)

- **Ipotesi**
 - Blocco dati di capienza 4 KB
 - *i-node* ampio 64 B
 - Indici di blocco espressi su 4 B
- **Esempio 1** (con uso di campo *single-indirect*)
 - Max dimensione di *file* rappresentabile
 - $(12 + 64 \text{ B} / 4 \text{ B}) \times 4 \text{ KB} = (12 + 16) \times 4 \text{ KB} = 112 \text{ KB}$
- **Esempio 2** (con uso di campo *double-indirect*)
 - Max dimensione di *file* rappresentabile
 - $112 \text{ KB} + 16^2 \times 4 \text{ KB} = 1 \text{ MB} + 112 \text{ KB}$
- **Esempio 3** (con uso di campo *triple-indirect*)
 - Max dimensione di *file* rappresentabile
 - $1 \text{ MB} + 112 \text{ KB} + 16^3 \times 4 \text{ KB} = 17 \text{ MB} + 112 \text{ KB}$

Realizzazione del FS in UNIX – 7

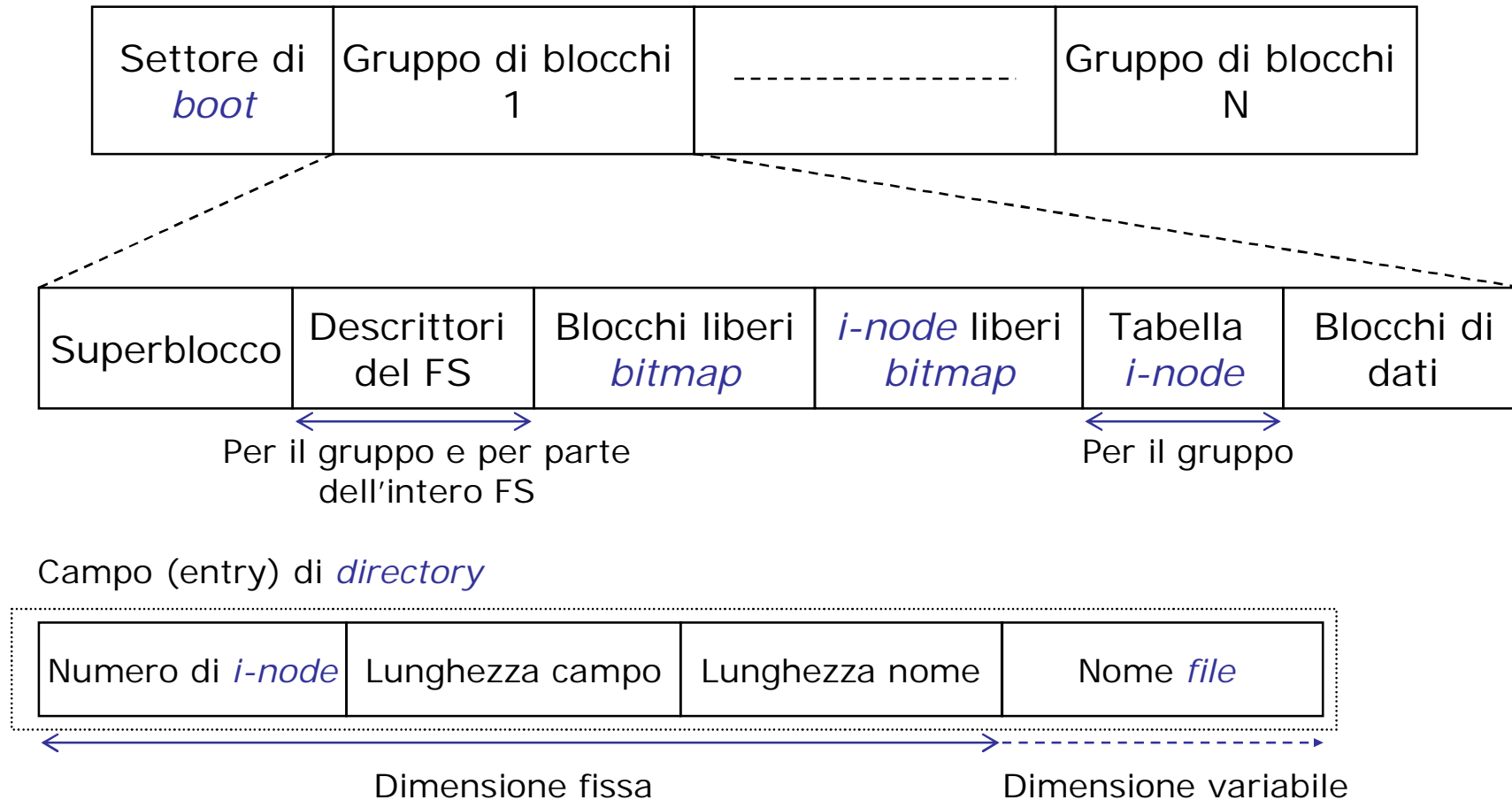
- La versione **BSD** introduce alcune migliorie importanti
 - Estensione del nome di *file* fino a 255 caratteri
 - *Directory* di dimensione **multipla** di blocco
 - Facilita e velocizza la scrittura su disco
 - Comporta frammentazione interna



Realizzazione del FS GNU/Linux – 1

- Inizialmente basato sul FS di MINIX però subito abbandonato per le eccessive limitazioni
 - Limitazioni imposte da MINIX
 - Nomi ≤ 14 caratteri
 - Indirizzi di blocco su 2 B per blocchi ampi 1 KB
 - Ampiezza massima di file ≤ 64 MB (**perché?**)
- **ext2** diviene presto la versione di riferimento
 - Basata sulle scelte BSD con diversa struttura fisica
 - La maggiore innovazione è stata la suddivisione della partizione in **gruppi di blocchi**
 - Distribuzione uniforme delle directory su disco
 - *i-node* e relativi blocchi dati sono tenuti **vicini** tramite preallocazione di alcuni blocchi al momento della creazione di un file
 - Riduce frammentazione

Realizzazione del FS GNU/Linux – 2



Frammentazione interna del blocco

Realizzazione del FS GNU/Linux – 3

- Dimensione di *i-node* estesa a 128 B
 - Indirizzi di blocco ampi 4 B
 - Per denotare fino a $2^{32} = 4$ G blocchi
 - Blocchi di dimensione 1, 2, 4 KB scelta in fase di configurazione del FS
 - Partizione di dimensione ≥ 4 TB
 - 12 indirizzi diretti + 3 indiretti (*single*, *double*, *triple*)
 - Informazioni di controllo
 - Una parte riservata per uso futuro
- Ogni aggiunta a *file* viene realizzata quanto più **localmente** possibile entro lo stesso gruppo
 - Località **tra** *file* correlati tramite gruppi
 - Località **entro** *file* mediante preallocazione di $N \leq 8$ blocchi contigui