

Linux Base - Capitolo n. 10

Edizioni ByteMan (15-12-2005)

revisione: 25/01/2008

XFree86

Il Sistema **X Window** è un ambiente grafico per sistemi UNIX di grande flessibilità e potenza, è stato sviluppato inizialmente al MIT; in seguito ad opera di alcuni sviluppatori commerciali è divenuto lo standard industriale per piattaforme UNIX.

XFree86 è invece una versione liberamente ridistribuibile del MIT X Window System versione 11, release 6 (X11R6) per sistemi UNIX 80386/80486/Pentium, include tutti i binari, i file di supporto, le librerie e tutti gli strumenti necessari.

L'ambiente grafico **X** è composto essenzialmente da tre parti:

1. **Windows manager** E' il gestore delle finestre, cioè fornisce un metodo per lavorare con le finestre. Esistono molteplici gestori: motif è lo standard commerciale più utilizzato ma i gestori più famosi utilizzati in ambiente Linux sono KDE, Gnome, Enlightenment, Sawfish, AfterStep, e FVWM.
2. **Server X** E' il processo che si occupa di gestire il display, ovvero si occupa di far interagire l'utente con la GUI. Esistono molteplici server X, ma su Linux il più utilizzato è di gran lunga XFree86, fornito gratuitamente. Alternative commerciali sono Accelerated-X e Metro-X.
3. **Client X** Sono tutti i programmi eseguiti sotto X, con cui l'utente interagisce.

KDE (K Desktop Environment) comprende oltre un windows manager un'intero ambiente desktop molto amichevole, ed inoltre fornisce vari sistemi integrati per la gestione e la configurazione del sistema.

Gnome (GNU Network Object Model Environment) è costituito, a differenza di KDE, solo dall'ambiente desktop per cui ha bisogno di un windows manager come FVWM (Fantastic Virtual Windows Manager). Anche Gnome offre dei propri sistemi integrati per la gestione della macchina. La sua architettura è completamente basata su CORBA.

Ogni distribuzione solitamente prevede la possibilità di installare ed utilizzare sia Gnome che KDE, è poi possibile modificarli e aggiornarli secondo le proprie necessità.

Il server **XFree86** è un eseguibile singolo binario (`/usr/X11R6/bin/XFree86`) che carica dinamicamente qualsiasi modulo X server necessario al momento dell'esecuzione dalla directory `/usr/X11R6/lib/modules/`. Alcuni di questi moduli sono montati automaticamente dal server, mentre altri sono facoltativi e devono essere specificati nel file di configurazione del server XFree86. Il file di configurazione del server XFree86 è `/etc/X11/XF86Config` che viene creato durante l'installazione di Linux mediante le informazioni raccolte nella fase iniziale dello stesso processo.

XF86Config

Dato che è molto raro dover modificare manualmente `XF86Config` è utile però conoscerne le varie sezioni ed i parametri di opzione disponibili, in modo particolare quando si effettua una messa a punto per malfunzionamenti.

Il file `/etc/X11/XF86Config` è costituito da tante piccole sezioni le quali indirizzano degli aspetti specifici dell'hardware del sistema.

Ogni sezione inizia con una riga **Section "section-name"** (dove section-name è il titolo della sezione) e finisce con una riga **EndSection**. All'interno di ogni sezione, ci sono le righe contenenti i nomi dell'opzione e almeno un valore dell'opzione, alcune volte riportato tra virgolette.

Le righe che iniziano con un carattere **#** non vengono lette dal server XFree86 e sono usate a scopo di commento.

Alcune opzioni all'interno del file accettano uno switch booleano che seleziona i contenuti **on** e **off**. I valori booleani accettati sono:

- 1, on, true, yes — Imposta l'opzione su on.
- 0, off, false, no — Imposta l'opzione su off.

Di seguito verranno riportate alcune delle sezioni piú importanti (Monitor, Device, Screen) visualizzate in un tipico file XF86Config. Per maggiori informazioni consultare le *man pages* relative alla voce XF86Config.

Ogni **sezione Monitor** configura un tipo di monitor usato dal sistema. Dato che una sezione Monitor é il minimo, ci possono essere casi con piú di una sezione monitor.

Il modo migliore di configurare un monitor é quello di configurarlo rispondendo alle domande durante il processo di installazione o usando lo Strumento di configurazione **X**.

Questo esempio riporta parte di una sezione Monitor tipica:

```
Section "Monitor"
    Identifier      "Monitor0"
    VendorName      "NEC"
    ModelName       "NEC3d68"
    DisplaySize     320 240
    HorizSync       31 - 69 # DDC-probed
    VertRefresh     55 - 120 # DDC-probed
    # 1024x768, 75.0Hz; hfreq=60.02, vfreq=75.03
    ModeLine "1024x768" 78.75 1024 1040 1136 1312 768 769 772 800 +hsync +vsync
EndSection
```

[Prestare la massima attenzione in caso di modifica manuale dei valori nella sezione Monitor di XF86Config. Valori inappropriati possono danneggiare seriamente il monitor. Consultare la documentazione per un elenco dei parametri operativi sicuri.](#)

Di seguito sono riportate le entry piú comuni usate nella sezione Monitor:

- **Identifier:** Specifica un nome unico per questa sezione Monitor. Questa é una entry necessaria.
- **VendorName:** Un parametro facoltativo che specifica il produttore del monitor.
- **ModelName:** Un parametro facoltativo che specifica il nome del modello del monitor.
- **DisplaySize:** Un parametro facoltativo che specifica in millimetri la misura fisica dell'area del monitor.
- **HorizSync:** Specifica la portata delle frequenze sync orizzontali compatibile con il monitor in kHz. Questi valori aiutano il server XFree86 e determinare la validitá delle entry Modeline interne o specificate per il monitor.
- **VertRefresh:** Specifica la portata delle frequenze di portata "refresh" supportate dal monitor in kHz. Questi valori aiutano il server XFree86 e determinare la validitá delle entry Modeline interne o specificate per il monitor.
- **Modeline:** Un parametro facoltativo che specifica modalitá di video aggiuntivi per il monitor, a risoluzioni particolari, con risoluzioni di refresh verticali e orizzontali. Consultare la pagina man XF86Config per una spiegazione piú dettagliata delle entry Modeline.

Ogni **sezione Device** configura una scheda video sul sistema. Una sezione Device é il minimo, ci possono essere istanze aggiuntive per ogni scheda video installata sulla macchina.

Il modo migliore di configurare un monitor é quello di configurarlo rispondendo alle domande durante il processo di installazione o usando lo Strumento di configurazione **X**.

Il seguente esempio illustra una sezione Device tipica per una scheda video:

```
Section "Device"
    Identifier      "Videocard0"
    Driver          "mga"
    VendorName      "Videocard vendor"
    BoardName       "Matrox Millennium G200"
```

```
VideoRam 8192
Option    "dpms"
EndSection
```

Di seguito sono riportate le entry più comuni usate nella sezione Device:

- **Identifier:** Specifica un nome unico per questa sezione Device. Questa é una entry necessaria.
- **Driver:** Specifica quale driver deve essere caricato dal server XFree86, in modo da utilizzare la scheda video. Un elenco dei driver puó essere trovato in /usr/X11R6/lib/X11/Cards, il quale viene installato con il pacchetto hwddata.
- **VendorName:** Un parametro facoltativo che specifica il produttore del monitor.
- **BoardName:** Un parametro facoltativo che specifica il nome della scheda video.
- **VideoRam:** Un parametro facoltativo che specifica la quantità di RAM disponibile sulla scheda video in kilobytes. Questa impostazione é solo necessaria per le schede video alle quali il server XFree86 non puó rilevare la quantità di RAM video.
- **BusID:** Una entry facoltativa che specifica la posizione del bus della scheda video. Questa opzione é solo necessaria per sistemi con schede multiple.
- **Option:** seguita da una stringa "<option-name>" che specifica parametri aggiuntivi per la sezione. Una delle opzioni piú comuni é "dpms", che attiva un "Service Star energy compliance" per il monitor.

Ogni **sezione Screen** collega una scheda video a un monitor, con riferimento alle sezioni Device e Monitor. Dato che una sezione Screen é il minimo, ci possono essere istanze aggiuntive per ogni combinazione scheda video e monitor presenti sulla macchina.

Il seguente esempio illustra una tipica sezione Screen:

```
Section "Screen"
    Identifier "Screen0"
    Device     "Videocard0"
    Monitor    "Monitor0"
    DefaultDepth 16
    SubSection "Display"
        Depth    24
        Modes    "1280x1024" "1280x960" "1152x864" "1024x768" "800x600" "640x480"
    EndSubSection
    SubSection "Display"
        Depth    16
        Modes    "1152x864" "1024x768" "800x600" "640x480"
    EndSubSection
EndSection
```

Le seguenti entry sono comunemente usate nella sezione Screen:

- **Identifier:** Specifica un nome unico per questa sezione Screen. Questa é una entry necessaria.
- **Device:** Specifica il nome unico di una sezione Device. Questa é una entry necessaria.
- **Monitor:** Specifica un nome unico di una sezione Monitor. Questa é una entry necessaria.
- **DefaultDepth:** Specifica l'intensità del colore in bit. Nell'esempio precedente, 16, che fissa in 65536 il numero di colori, é il default. Entry DefaultDepth multiple sono permesse, ma deve esserne presente almeno una.
- **SubSection "Display":** Specifica le modalità della schermata disponibili con una particolare intensità di colore. Una sezione Screen puó avere sottosezioni Display multiple, ma ci deve essere almeno una sezione per la profondità del colore specificato nella entry DefaultDepth.

/proc (filesystem)

Il file system **/proc** e' un file system creato e mantenuto a run-time dal Kernel di Linux per tenere traccia dei vari processi che stanno funzionando sulla macchina e del loro stato. Questa gerarchia di file speciali, che rappresentano lo stato corrente del kernel, consente alle applicazioni e agli utenti di esplorare il sistema attraverso il punto di vista del kernel. Tramite **/proc** si possono ottenere tantissime informazioni su ciò che e' installato, su come e' configurato e su come funziona. La directory /proc contiene

Una delle cose da tenere presente è che **/proc** in realtà non esiste, ma viene creata dinamicamente nella memoria del sistema in *tempo reale*; **/proc** puo' essere letta da qualunque utente abbia i corretti permessi di accesso, alcune delle sue parti pero' possono essere lette solamente dal proprietario del processo in questione e, ovviamente, da root.

Sotto linux, vi sono varie utility come **lscpi**, **scanpci**, ed altre che ottengono i loro dati proprio leggendo **/proc** ed interpretandone le informazioni. Altre utility simili sono ps e top.

La root (/) di un sistema Linux di solito si presenta così:

```
.....
drwxr-xr-x  20 root root   480 Dec 16 20:54 mnt
drwxr-xr-x   2 root root    48 Mar  7  2005 opt
dr-xr-xr-x 112 root root     0 Dec 18 06:28 proc <---
drwxr-xr-x  22 root root   960 Dec 18 06:11 root
drwxr-xr-x   2 root root  5136 Dec 15 19:13 sbin
drwxr-xr-x  10 root root     0 Dec 18 06:28 sys
.....
```

Si noti che poichè **/proc** e' un file system virtuale, esso viene ricreato ogni volta che la macchina viene accesa, e quindi la sua data/ora e' sempre aggiornata. Il listato del suo contenuto assomiglia a qualcosa del genere:

```
1, 105, 1353, 1354, 16, 160, 161, 162, 163, 2, 2671, 3, 3431, 4, 4262, 4265,
4270, 4353, 4423, 4424, 4425, 4426, 4427, 4505, 4511, 4516, 4518, 4525, 4530,
4531, 4535, 4536, 4537, 4538, 4539, 4540, 4557, 4563, 4564, 4565, 4566, 4567,
4623, 4624, 4643, 4681, 4708, 4711, 4713, 4716, 4723, 4729, 4747, 4749, 4752,
4763, 4765, 4767, 4770, 4784, 4797, 4800, 4801, 4809, 4811, 4813, 4946, 4954,
4963, 4967, 4969, 4971, 4976, 4979, 4991, 4999, 5002, 5003, 5004, 5011, 5013,
5090, 754, 839, 844, 845, 857, 882, 883, 886, 92, acpi, asound, buddyinfo,
bus, cmdline, cpuinfo, crypto, devices, diskstats, dma, driver, execdomains,
fb, filesystems, fs, ide, interrupts, iomem, ioports, irq, kallsyms, kcore,
kmsg, loadavg, locks, meminfo, misc, modules, mounts, mtrr, net, partitions,
pci, scsi, self, slabinfo, splash, stat, swaps, sys, sysrq-trigger, sysvipc,
tty, uptime, version, vmstat
```

Si notino una serie di directory contrassegnate da un numero, si tratta dei singoli processi in esecuzione nella macchina nell'istante in cui è stato eseguito il comando `ls -m /proc`. Per "vedere" il contenuto di ogni singola directory occorre essere root, questo perché se il processo appartiene ad un diverso utente ne viene proibito l'accesso. A titolo di esempio osserviamo il contenuto di una di esse:

```
totale 0
dr-xr-xr-x   3 root root 0 2005-12-19 09:35 .
dr-xr-xr-x  90 root root 0 2005-12-19 09:35 ..
dr-xr-xr-x   2 root root 0 2005-12-19 09:56 attr
-r-----   1 root root 0 2005-12-19 09:56 auxv
-r--r--r--   1 root root 0 2005-12-19 09:56 cmdline
lrwxrwxrwx   1 root root 0 2005-12-19 09:56 cwd -> /
-r-----   1 root root 0 2005-12-19 09:56 environ
lrwxrwxrwx   1 root root 0 2005-12-19 09:56 exe
dr-x-----   2 root root 0 2005-12-19 09:56 fd
-r--r--r--   1 root root 0 2005-12-19 09:56 maps
-rw-----   1 root root 0 2005-12-19 09:56 mem
-r--r--r--   1 root root 0 2005-12-19 09:56 mounts
-rw-r--r--   1 root root 0 2005-12-19 09:56 oom_adj
-r--r--r--   1 root root 0 2005-12-19 09:56 oom_score
lrwxrwxrwx   1 root root 0 2005-12-19 09:56 root -> /
```

```
-r--r--r-- 1 root root 0 2005-12-19 09:41 stat
-r--r--r-- 1 root root 0 2005-12-19 09:56 statm
-r--r--r-- 1 root root 0 2005-12-19 09:41 status
dr-xr-xr-x 3 root root 0 2005-12-19 09:56 task
-r--r--r-- 1 root root 0 2005-12-19 09:56 wchan
```

Il contenuto delle altre directory numeriche e' praticamente lo stesso, indipendentemente dalla directory che scegliamo. Solo i valori dei singoli parametri cambiano a seconda del processo scelto. Dall'esame di alcuni file, contenuti all'interno di una sottodirectory di /proc che identifica un processo, si possono scoprire delle informazioni interessanti relative al processo stesso:

```
-r--r--r-- 1 root root 0 2005-12-19 09:56 cmdline
File contenente l'intera linea di comando invocata per avviare
il processo, non viene usata nessuna formattazione particolare.
```

```
-r----- 1 root root 0 2005-12-19 09:56 environ
File contenente tutte le variabili di ambiente definite per quel
particolare processo, nella forma VARIABILE=valore,
non viene usata nessuna formattazione particolare.
```

```
lrwxrwxrwx 1 root root 0 2005-12-19 09:56 root -> /
Questo link mostra quale e' la directory "root" per quel particolare
processo. Un processo può, infatti, avere una root che non corrisponde
alla "vera" root.
```

```
-r--r--r-- 1 root root 0 2005-12-19 09:41 status
File che fornisce informazioni relative allo stato del processo,
(dormiente, attivo,...), PID, UID, PPID ed altre informazioni simili.
Queste informazioni possono essere visualizzate usando ps e top.
```

Sinteticamente ecco le informazioni riportate da alcuni di questi file:

/proc/cpuinfo Riporta informazioni sul processore, per esempio il tipo, marca, modello e le varie performance.

/proc/devices Elenca i driver configurati nel kernel corrente

/proc/dma Mostra quali canali DMA sono in uso al momento

/proc/filesystems Mostra i filesystem configurati (supportati) nel kernel

/proc/interrupts Visualizza gli interrupts in uso e quanti e quali sono stati usati.

/proc/ioports Mostra quali porte di i/o sono in uso al momento

Applicativi (parte 4)

Computer oggi è sinonimo di Internet. Quali sono dunque i principali browser che vengono utilizzati in ambiente Linux? A questo interrogativo cercherà di rispondere questa sezione nella quale verranno passati in rassegna alcuni dei browser più diffusi nel mondo del Pinguino.

Mozilla Firefox

È un browser gratuito, naturalmente, e open-source disponibile per Linux, Windows, e MacOS X. È leggero, veloce, facile da usare e offre numerosi vantaggi rispetto agli altri browser, come la navigazione per schede e la possibilità di bloccare le finestre pop-up.

La navigazione a schede modifica il proprio modo di navigare in Internet, permettendo l'apertura dei collegamenti di una pagina in un'unica finestra del browser, raccolti in schede che rimangono dietro la scheda corrente, disponibili per una lettura successiva. Con la navigazione a schede è più semplice visualizzare più pagine o siti contemporaneamente, raggruppandoli in una sola finestra. Mentre si sta leggendo una pagina, le pagine aperte in schede si caricano completamente così da essere pronte per la lettura senza ulteriori attese.



Mozilla Firefox blocca efficacemente le finestre pop-up non richieste (pubblicità, ed altro) e mostra invece quelle desiderate con un semplice meccanismo a lista bianca.

Firefox è il primo browser con la possibilità di mantenersi aggiornato automaticamente. Si collega, infatti, al sito di Mozilla update, ed informa l'utente della presenza di eventuali aggiornamenti per il software in uso. L'Aggiornamento software si preoccupa anche di verificare se sono disponibili aggiornamenti per le estensioni installate. Firefox è in grado di ricercare gli aggiornamenti mentre si sta navigando ed informare l'utente con un avviso, se ne risultano disponibili.

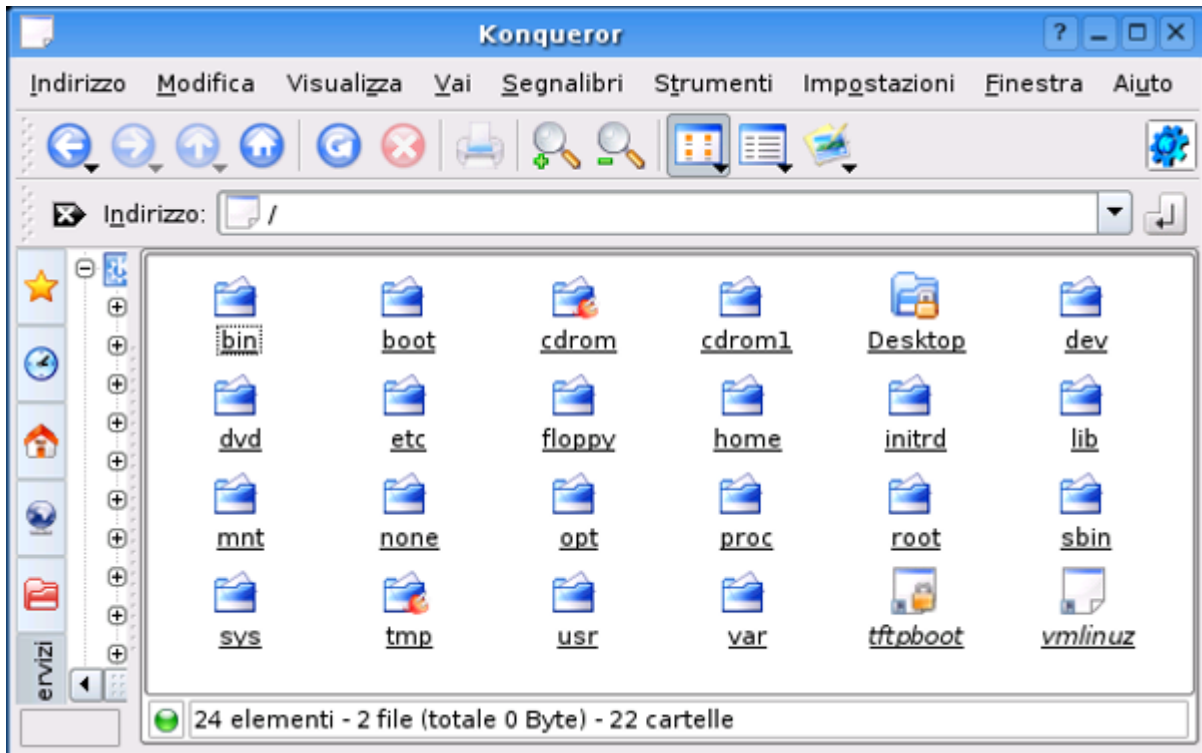
Konqueror

È certamente qualcosa in più di un semplice browser, è anche un gestore di file ed un visualizzatore universale di documenti. Integrato nell'ambiente grafico KDE, è disponibile per diversi sistemi operativi Unix e Linux, ma non è scaricabile o installabile separatamente da KDE. Fra le sue caratteristiche principali si segnala la capacità di visualizzazione e, quando possibile, di stampa di un vasto numero di formati di file.

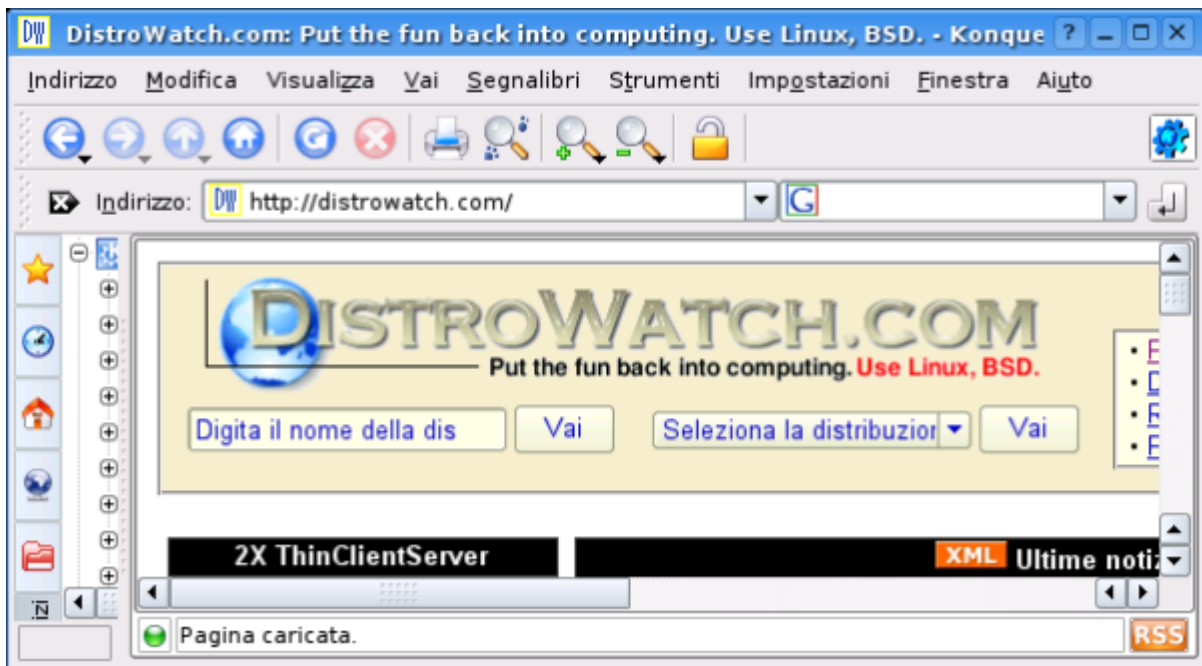
Per la navigazione Konqueror poggia sul componente che gestisce la visualizzazione delle pagine HTML all'interno del progetto KDE, vale a dire KHTML. KHTML è anche alla base del browser ufficiale di Mac OS X, Safari.

Esiste ancora un progetto parallelo al Browser principale, Konqueror/Embedded che punta a

produrre una versione dell'applicativo utilizzabile su dispositivi embedded, come palmari o altri dispositivi portatili.



L'interfaccia utente è gradevole, completa e personalizzabile. È basata su pannelli che possono essere spostati e aggiunti. Per esempio è possibile avere la lista dei siti preferiti nel pannello a sinistra della finestra di navigazione, e cliccando sul segnalibro, quest'ultimo si apre nel pannello di destra. Nello stesso modo, è possibile visualizzare l'albero delle cartelle e delle periferiche a sinistra e i file contenuti a destra.



I pannelli sono abbastanza flessibili e possono persino includere una finestra con una console integrata. La configurazione dei pannelli può essere salvata (è anche possibile fare la stessa cosa con i tasti di scelta rapida, anch'essi modificabili), inoltre sono presenti delle configurazioni predefinite (come Midnight Commander che divide lo schermo in due parti e ognuna di esse mostra pagine

Web, cartelle, file, ecc). È possibile utilizzare tutte le funzioni di navigazione (indietro, avanti, cronologia, ecc) in tutte le operazioni. Infine, la barra degli indirizzi autocompleta gli indirizzi dei file locali e supporta gli alias per gli URL esterni e le funzioni di ricerca.

Altri browser

Oltre i due citati precedentemente, esistono altri browser in ambiente Linux, vale la pena ricordare i seguenti:

- **Galeon**, funziona col desktop Gnome e si appoggia a Mozilla. Nasce dall'idea di volere un'applicazione *solo per navigare*, si è voluto sviluppare un programma per la navigazione che non risultasse appesantito da tutta una serie di applicazioni aggiuntive e che servisse semplicemente da browser.
- **Nautilus**, funziona anche come file manager nel desktop Gnome e si appoggia a Mozilla.
- **Opera**, celebre browser e client di posta, supporta tutte le maggiori distribuzioni di Linux. Supporta bene le caratteristiche standard dei moderni siti web. E' leggero e compatto ed efficiente.
- **Lynx** e **Link** sono due progetti con origine comune. Entrambi permettono la visione delle pagine web anche su una console testuale. Non supportano invece le immagini, gli applet java, i javascript e in generale gli script eseguiti da lato client. Si tratta quindi di programmi spartani, ma molto leggeri e veloci (visto che non vengono appesantiti dalla grafica) e facilmente utilizzabili una volta presa confidenza con le sequenze di tasti. Permettono di accedere a siti ftp e di fare il download di file.