



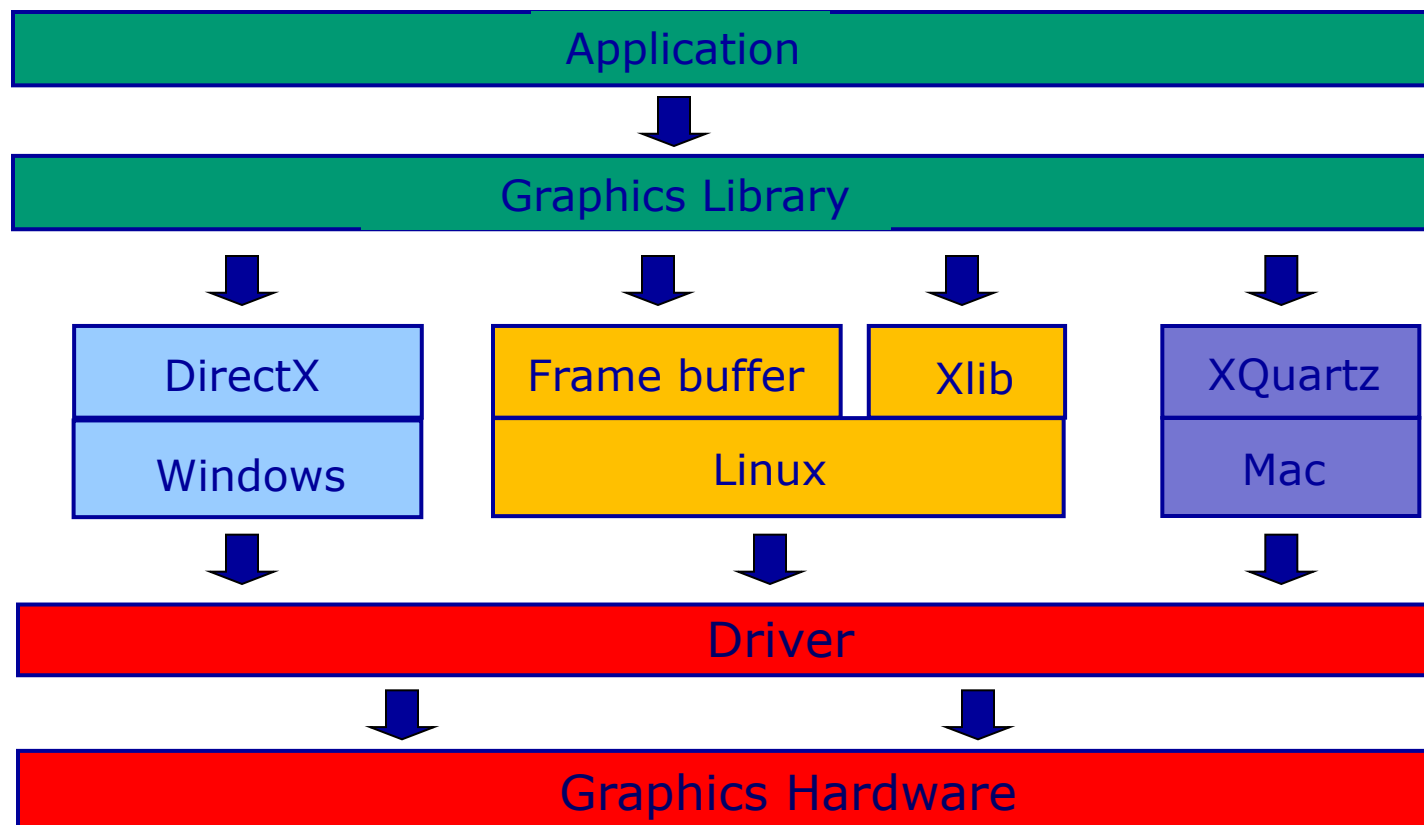
# Libreria Grafica XWindow



<https://www.x.org/wiki/>



# Struttura di un'Applicazione Grafica



Quartz: mac Core Graphics framework

DirectX 12: microsoft windows 10



# X Window System

---

Il progetto X Window System ebbe origine al Massachusetts Institute of Technology (MIT) nel 1984.

La versione corrente di protocollo, X11, apparve nel settembre del 1987.

Dal 2004 **X.Org Foundation** guida il progetto con l'implementazione corrente X.Org Server, disponibile come "free and open source software" sotto licenza MIT.





# XOrg

Nel 2005 è stato rilasciato XOrg X11R7.0, il primo grande rilascio del sistema X Window dopo dieci anni di sviluppo discontinuo. Oltre a driver video aggiornati, troviamo DRI (Direct Rendering Infrastructure) per creare implementazioni di OpenGL che sfruttino l'accelerazione grafica 3D.

DRI è una parte integrale di X.Org 7.x, ed è integrata con Mesa, una implementazione open source delle API OpenGL. Parecchi driver per schede accelerate 3D sono stati scritti con le specifiche DRI, fra cui ATI, Matrox, 3DFX e Intel.

Nel giugno 2012 è stato rilasciato XOrg X11R7.7 che è a tutt'ora l'ultima versione rilasciata (in verità l'ultima è la X11R7.8 con extension aggiuntive rispetto alla X11R7.7).



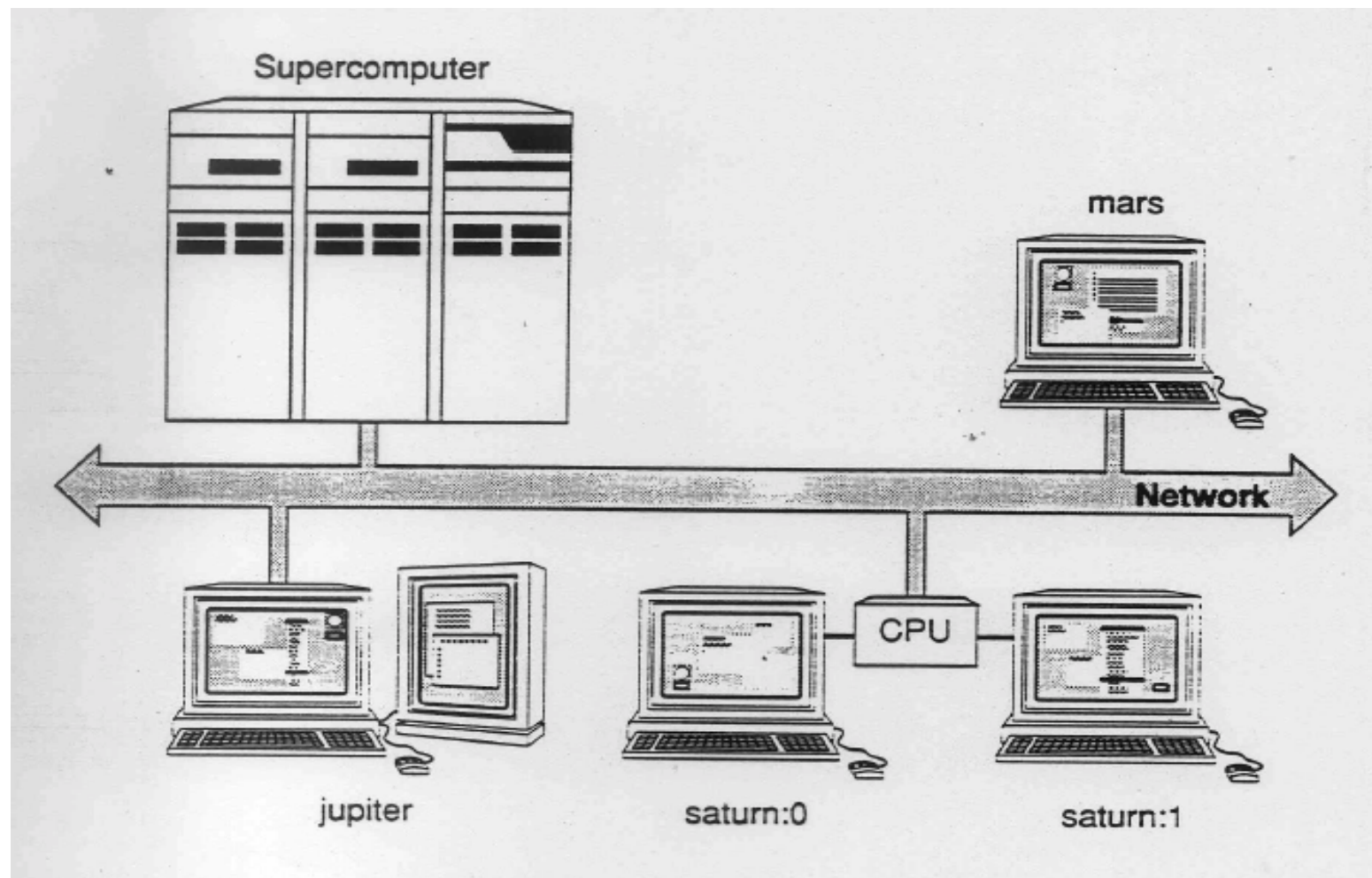
# Il Sistema X Window

- Indipendente da HW e S.O., permette di realizzare applicazioni portabili su macchine diverse sotto S.O. diversi
- Progettato per **sistemi distribuiti**, permette a più applicazioni di cooperare attraverso una rete anche con architetture e/o S.O. diversi. Tutto avviene in maniera completamente trasparente all'applicazione.

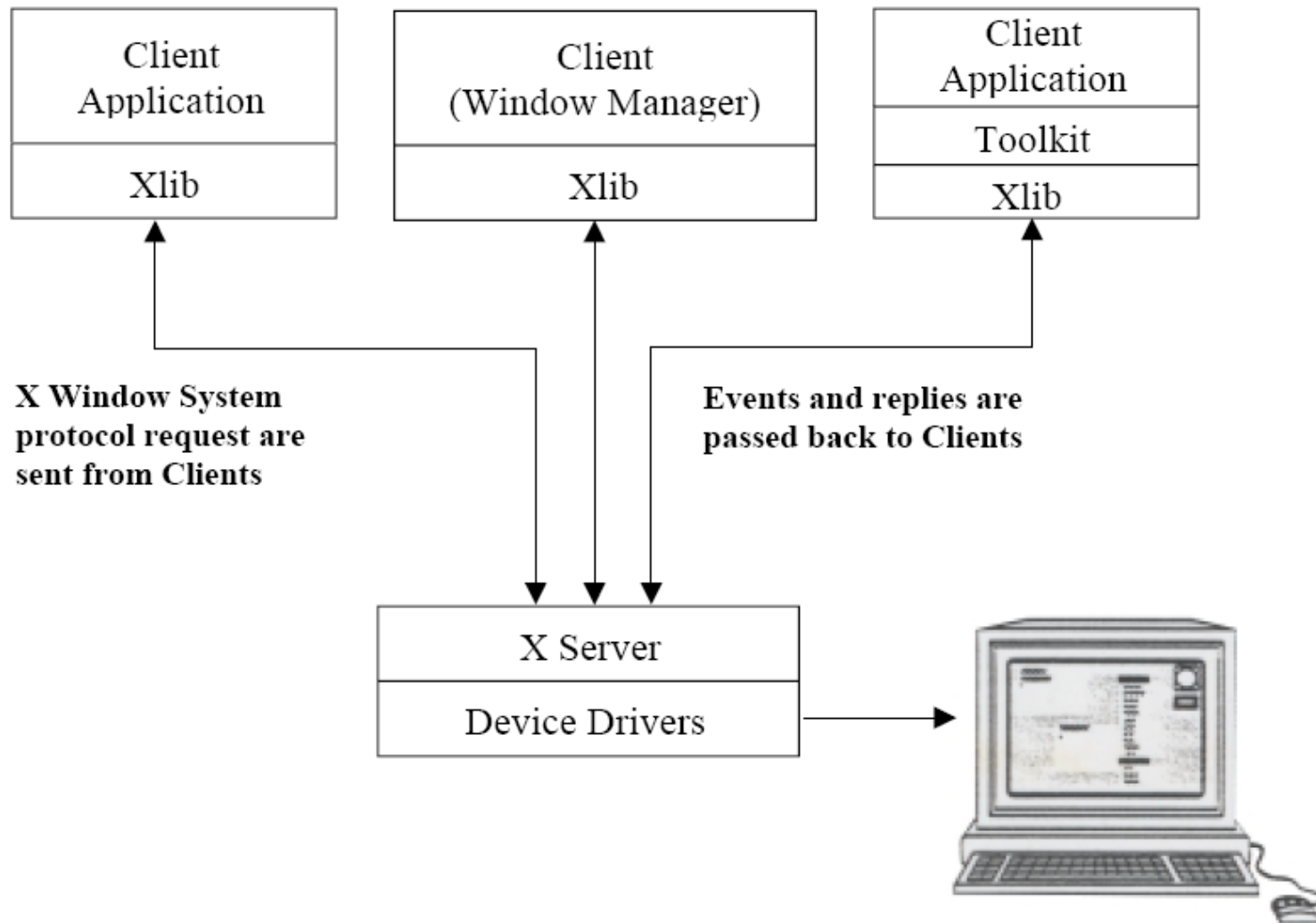
Nota: l'interfaccia grafica (o Window Manager) può girare su una macchina diversa da quella dove gira l'applicazione.

- X non impone alcun stile di interfaccia (o Window Manager) o di interazione con l'utente. Questo viene definito ad uno strato software sopra ad X

# Il Sistema X Window

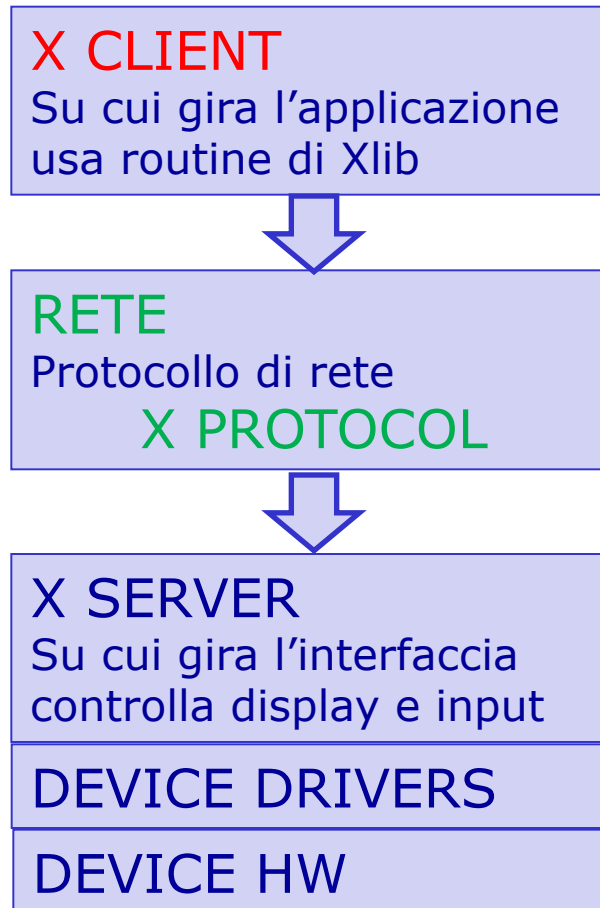


# Il Sistema X Window





# Il Sistema X Window







# X Protocol

Definisce quali pacchetti di informazioni (messaggi) sono scambiati tra **Client** e **Server** tramite **Rete**:

## **RICHIESTE** (**Client** → **Server**)

Alloca una risorsa, legge valori correnti di una risorsa, modifica valori di una risorsa. Interazione asincrona: il server mette le richieste in coda e le soddisfa quando ha tempo.

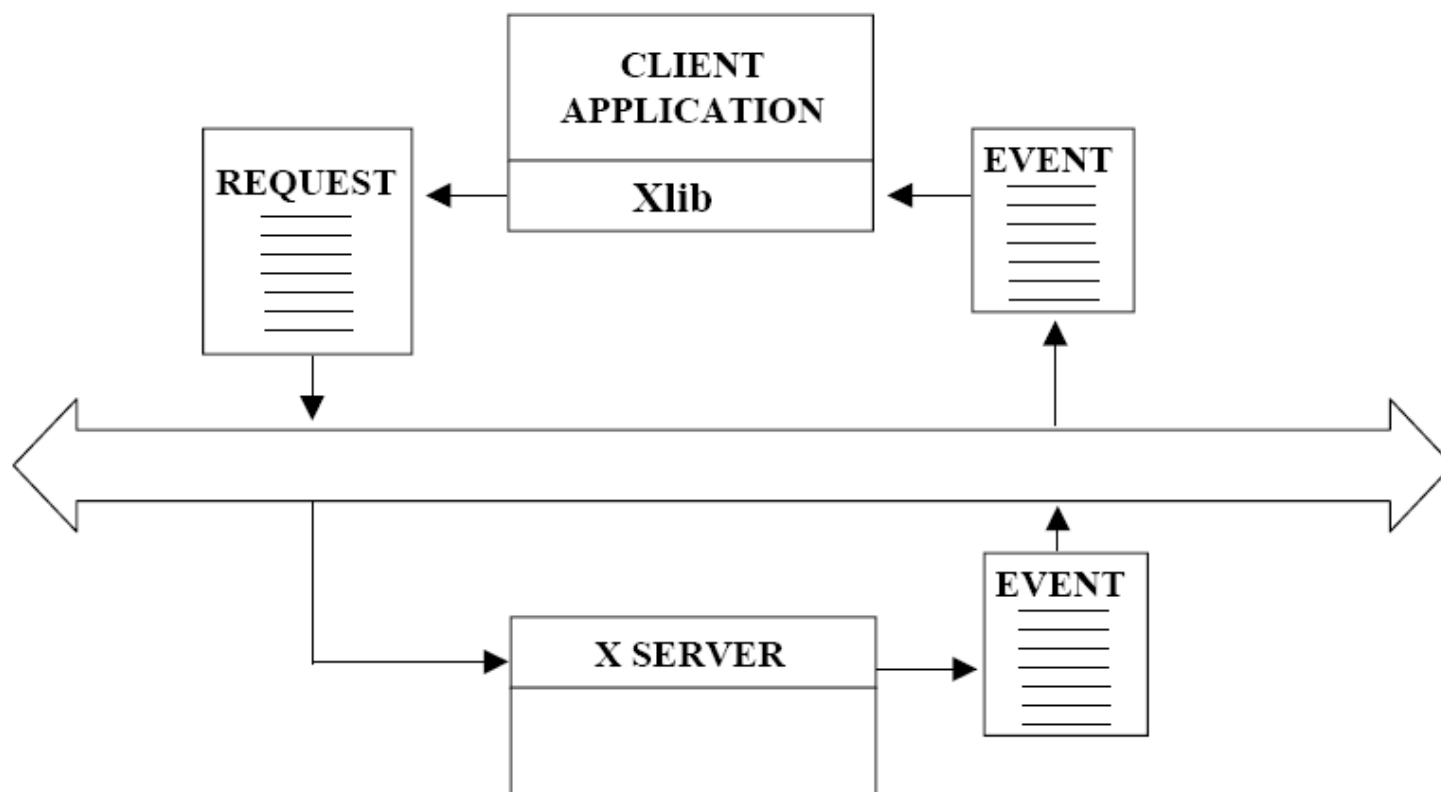
## **RISPOSTE** (**Server** → **Client**)

Info di ritorno per certi tipi di richieste (es. alloca una risorsa, ritorna identificatore per accedere alla risorsa).

## **EVENTI** (**Server** → **Client**)

Xlib (sul client) mette eventi in coda da cui il programma può leggerli uno ad uno per processarli (interazione asincrona)

# X Protocol





# X Server e X Client

## X Server

- Alloca e gestisce risorse (finestre, font, bitmap, pixmap, ...) per conto del client: solo l' **id**entificatore della risorsa è noto al client e passato via rete (insieme ai messaggi che si riferiscono a quella risorsa)
- Passa input dai device (tastiera, mouse) ai client
- Passa output (testo e grafica 2D) dai client al display

## X Client

Un'applicazione che usa un X Server tramite Xlib.



# Xlib

Strato più basso di X. Permette al client di comunicare col server per:

- Ottenere risorse (font, pixmap, finestre, ...)
- Manipolare risorse (leggere, assegnare e modificare proprietà)
- Ricevere input (eventi)
- Inviare output (testi e grafica 2D)



# Connessione con ssh ad un X server remoto

`$ssh -X loginname@remotehost.domain`

quindi

Password: ...

L'uso di ssh garantisce una connessione sicura da un  
X Server locale ad una applicazione remota

su Mac dare

`$ssh -Y loginname@remotehost.domain`



# Connessione con ssh ad un X server remoto

In che ordine si deve procedere?

1. Lanciare l'X Server sull'host locale
2. Aprire una shell (xterm) sull'host locale
3. Eseguire ssh per stabilire una connessione con la macchina remota (comandi visti prima)
4. Dare comandi X sulla macchina remota

Questo permette la visualizzazione dell'output del XClient remoto come se fosse sulla macchina locale

facciamo una prova di grafica remota?



ALMA MATER STUDIORUM  
UNIVERSITÀ DI BOLOGNA

**Giulio Casciola**  
Dip. di Matematica  
[giulio.casciola@unibo.it](mailto:giulio.casciola@unibo.it)