

tesi di laurea

Sistema Operativo Chrome: Analisi degli aspetti peculiari.

Anno Accademico 2009/2010

relatore

Ch.mo prof. Porfirio Tramontana

candidato

Lina Cocomello

Matr. 534/000565

Obiettivi.

- **Che cos'è Chrome OS?**
- **Come è strutturato?**
- **In che modo viene gestito lo sviluppo del software?**
- **Come si interfaccia con il Web?**
- **Come si comporta con le Applicazioni Web?**
- **Confronti e Conclusioni.**



Google Chrome OS

Google Chrome OS.

- **Sistema Operativo Web.**
- **È basato sul kernel Linux.**
- **Il codice del S.O. (Chromium OS) e del Browser (Chromium Browser) è open-source ed ancora in fase di sviluppo, quindi ampliabile dalla community di sviluppatori.**
- **Target di Google: utente medio che utilizza il PC prevalentemente per il Web.**
- **Dispositivi secondari (Netbook, Tablet, ecc.) ottimizzati nell'HW dai produttori per vendere "macchina Chrome"**
→ **buon rapporto qualità/prezzo.**
- **Obiettivi di Google:**
Velocità, Sicurezza, Semplicità,
Affidabilità, Portabilità.

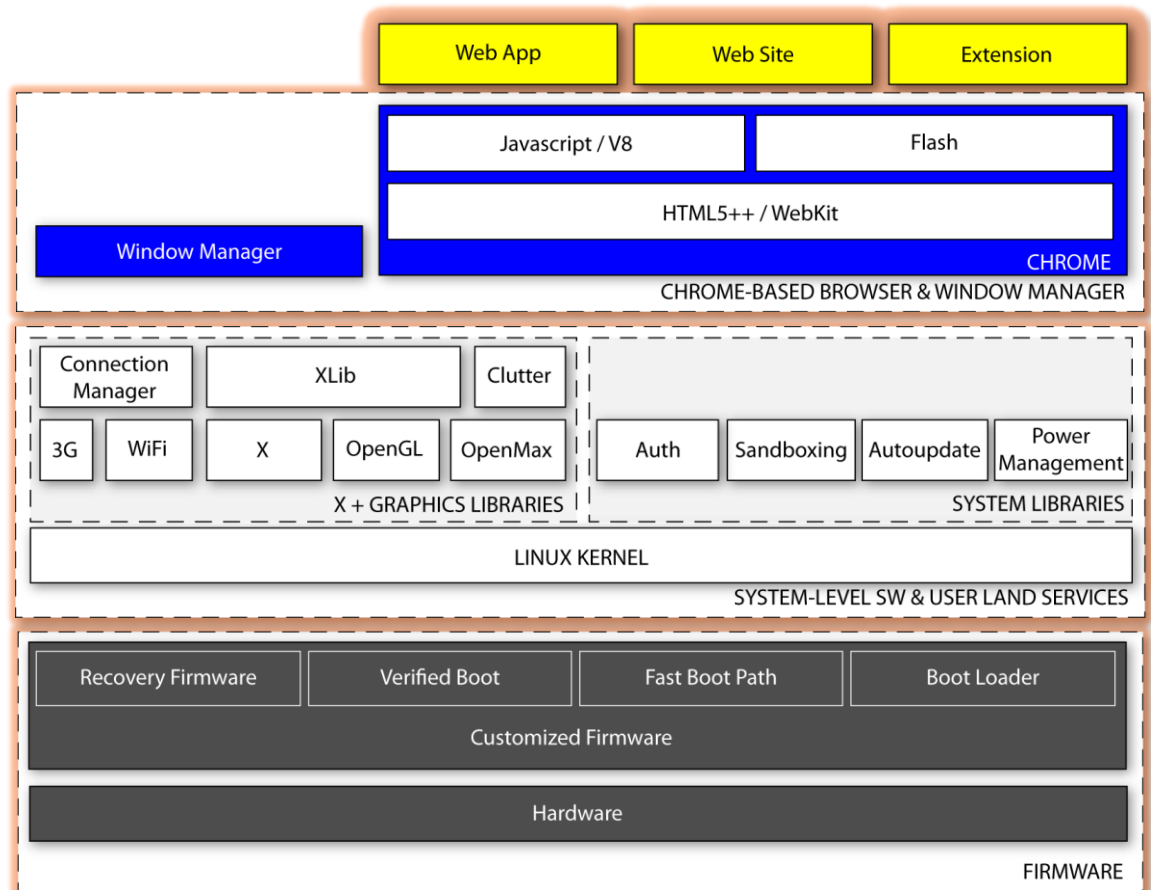


Chrome OS - Architettura.

Come è progettato?

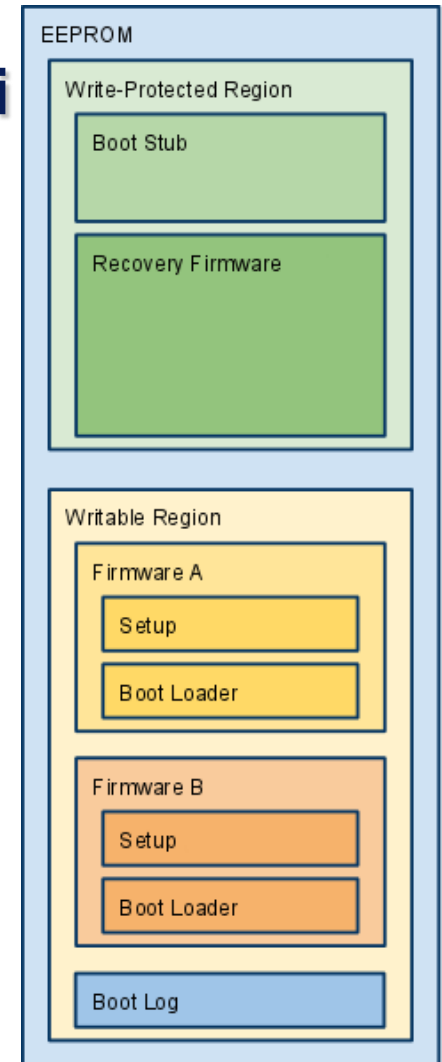
Architettura a strati:

- Firmware,
- System-level SW e Demoni,
- Chrome-based Browser e Window Manager.



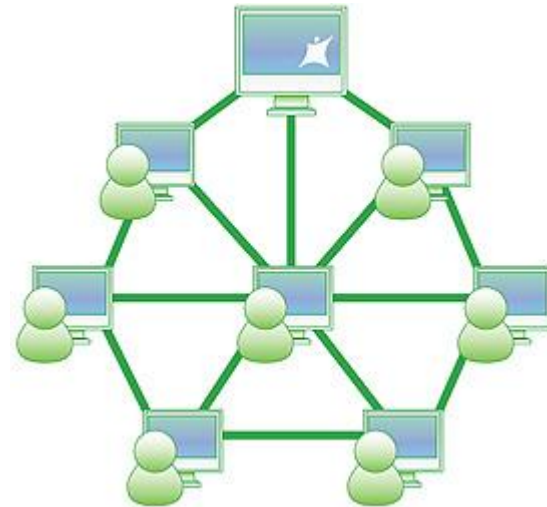
Architettura - Aspetti peculiari

- **Avvio verificato.**
- **Ripristino del sistema.**
- **File system.**
 - Ext4
- **Protezione dei dati.**
 - Crittografia in remoto (file system, dati utente)
 - Immagine della home directory crittografata in locale



Chromium: Ambiente di sviluppo dei progetti.

- **SCM distribuito.**
 - **Approccio peer-to-peer.**
 - developer-client per il pull
 - developer-server per il push
 - ogni pari è un archivio
 - scambio di patch tra pari = sincronizzazione
 - **Struttura come file system.**
 - Ricerca e visualizzazione veloce
 - Ogni cartella ha storia e tracciatura completa
- **Stile di programmazione.**



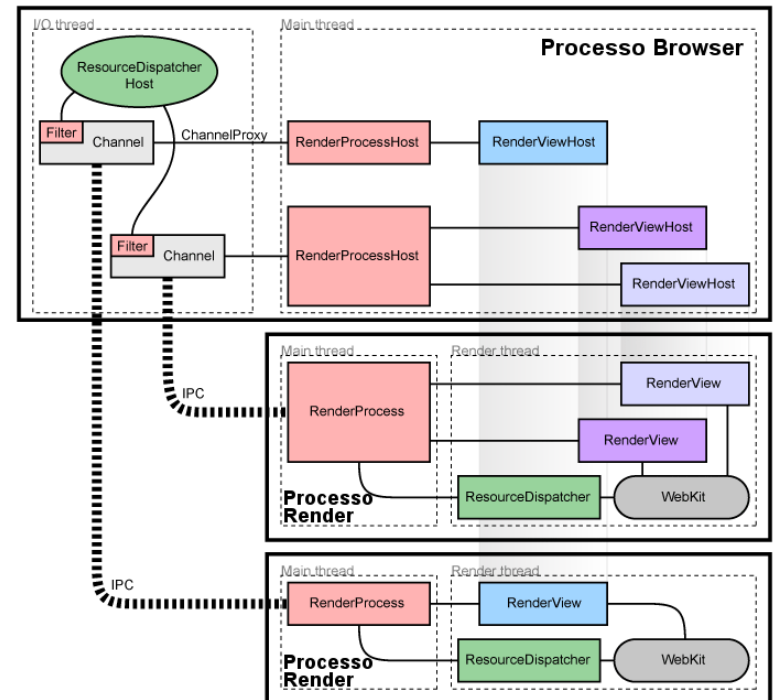
Chrome Browser.

■ Architettura multi-processo.

- Un processo per ogni tab → sicurezza (sandboxing) e affidabilità (no browser crash)
- Processo Browser (padre) esegue GUI.
- Processi Render (figli) con pochi privilegi.

■ IPC (Inter-Process Communication):

- Primitiva pipe per I/O tra i processi
- In modalità asincrona.



Chrome: Web Application.

- Implementate con NativeClient, quindi potenziate da librerie di codice nativo (es. C/C++)
- Web Apps:
 - online,
 - always offline,
 - online-offline.
- Storage lato Client e supporto di HTML5.
- Chrome Web Store:
 - vetrina di Web Apps.



Chrome

Vs.

Altri S.O.

■ Chrome WebOS.

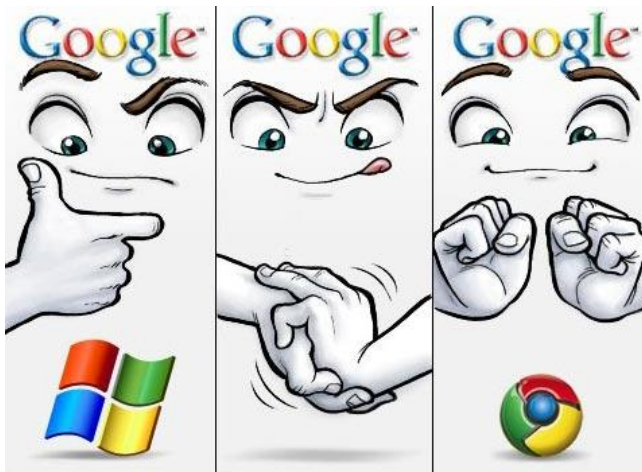
- Open source Web OS, portabile, economico (“macchina Chrome”).
- Avvio veloce (27 s), semplice, sicuro (al riavvio, dati, filesystem, home-directory).
- Supporto SSD per SO: occupa solo 1/60 rispetto a Windows 7.
- Utilizzo completo dovunque e comunque (web apps disponibili, storage on demand, cloud printing, ...)

■ Windows 7

- Desktop OS, proprietario, non portabile, caro.
- Avvio ridondante (55 s), non sempre intuitivo, password per sessione.
- Richiede 20 GB di spazio su HDD per architettura a 64 bit.
- Utilizzo completo solo dalla workstation, uso Web apps non ottimale.

■ Linux (Ubuntu 9.10)

- Open-source Desktop OS, portabile, gratuito.
- Avvio ridondante (50 s), non sempre intuitivo ma personalizzabile, sicuro.
- Richiede 8 GB di spazio su HDD per architettura a 64 bit.
- Utilizzo completo dalla workstation, uso Web apps (anche storage on demand, ...).



Confronti e Conclusioni.

Chrome

Vs.

Centralized SCM/Altri Browser/ Desktop App

■ SCM distribuito: GIT.

- Approccio peer-to-peer.
- Struttura ad albero \approx filesystem

■ Chrome Browser.

- Veloce, Affidabile, Sicuro, supporto HTML5.

■ Web Apps.

- Ad oggi varietà sufficiente.
- Economiche.
- Difficile uso di graphic App, video editing.

■ SCM centralizzato.

- Server centralizzato \rightarrow
- Connessione lenta per:
 - Caricare copia,
 - Ricerca elemento
 - Upload modifiche

■ Browser IE.

- Non veloce, Affidabile, Sicuro.

■ Firefox Mozilla.

- Veloce, poco affidabile, poco sicuro.

Applicazioni desktop.

- Varietà immensa.
- Non economiche.
- Anche uso graphic App, video editing che richiede ampio uso di memoria.

