



*Siamo a un punto di svolta in questo settore tra standard da rinnovare, nuovi controlli di utilizzo dei contenuti e problemi di interoperabilità.*

## COME CAMBIANO I BROWSER

Il desiderio di usare Internet *anche* come canale televisivo, o come videoregistratore collegato a un unico archivio universale, è nato insieme al Web. Una delle sue conseguenze più importanti, a livello software, è l'ubiquità della tecnologia Flash di Adobe, che è ancora l'*unico* sistema (almeno nel Web) con cui moltissimi siti distribuiscono video. Questo monopolio di fatto, già incrinatosi quando Apple decise di non supportare Flash nei suoi dispositivi mobili, ha subito di recente nuovi e seri attacchi.

All'inizio del 2015, per esempio, i browser degli utenti Linux, e non solo, hanno iniziato (o ricominciato...) a riempirsi di pressanti sollecitazioni ad aggiornare Flash. Il motivo era la pericolosità delle versioni precedenti, e la scoperta che le vulnerabilità erano state usate per attacchi informatici anche attraverso normali banner pubblicitari, o video caricati su portali normalmente affidabili, come DailyMotion.

Queste continue vulnerabilità di Flash hanno contribuito alla decisione di YouTube di servire nuovi video esclusivamente con tecnologia Html5, e

## Che futuro ha l'Open Source nel mondo del video online?

**I**l 2015 potrebbe essere l'anno in cui diversi nodi legati alla (auto)produzione, distribuzione e consumo di video digitali, nati negli ultimi due o tre anni, verranno al pettine. Quali sono i principali problemi che il software Open Source e i formati aperti possono aiutare a risolvere? Lo scopriremo in queste pagine, presentando anche, agli utenti più intraprendenti, alcune delle possibilità Open Source di streaming video.

### LO SPETTRO DI BREVETTI E DRM

Il video in qualunque forma, dai micro filmati fatti o visti col telefonino ai kolossal di Hollywood, muove o tocca enormi interessi economici. La conseguenza più ovvia, per quanto concerne questa rubrica, è che quasi tutte le tecnologie per produrre video di qualità e distribuirlo, dai cinema ai tablet passando per i lettori di dischi, sono piene di brevetti da un lato e sistemi di protezione dall'altro.

Questi ultimi sono chiamati Dm (*Digital Rights Management*), cioè gestione dei diritti digitali, o gestione digitale dei vari diritti legati alla proprietà intellettuale. Negli ultimi anni il loro campo d'azione si è progressivamente

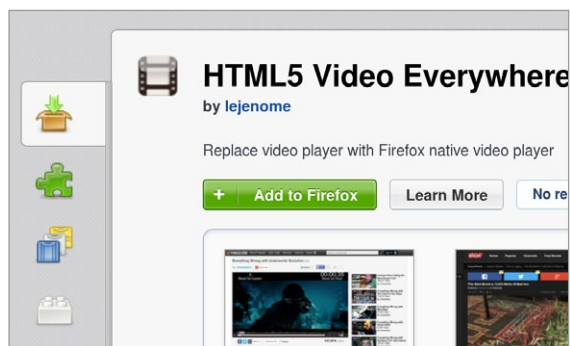
esteso, dai primissimi e poco sicuri sistemi anticopia di cassette Vhs e Dvd ai vari meccanismi attualmente in utilizzo per impedire la copia su computer di video dal vivo ricevuto via Internet.

Normalmente si discute di questi sistemi solo in relazione alla cosiddetta *pirateria*, ovvero alla copia non autorizzata, privata o per fini di lucro, di materiale coperto da diritto d'autore. In realtà, qualsiasi opinione si abbia sul copyright, parlare di formati video *soltanto* a quel livello o a quello della pura qualità delle immagini sarebbe molto riduttivo.

Un mondo video composto *esclusivamente* di sistemi chiusi e proprietari potrebbe creare seri problemi sia alla *produzione* di video originali, sia in generale alla sicurezza e alla usabilità di Internet.



L'ennesima richiesta di aggiornare Flash per seri motivi di sicurezza: riusciremo nel 2015 a superare questa fase della storia dei browser?



**La promessa di Firefox (e qualsiasi altro browser) per il prossimo futuro: pochi player video nativi, anziché stuoli di plugin più o meno compatibili. Ma c'è il trucco...**

dato l'avvio a suggerimenti di disinstallare Flash senza troppi patemi d'animo, "tanto ormai non serve più". In parallelo a queste disavventure di Flash, Netflix, Bbc e altri grandi produttori o distributori video hanno aumentato le pressioni sugli sviluppatori di browser, per bloccare almeno in ambito video certe possibilità che tutti ormai diamo per scontate, fin dalla nascita del Web: il caso più comune e importante è la possibilità di salvare sul proprio computer, checché ne pensino in proposito l'autore o il distributore di quei contenuti, una copia perfetta di quanto stiamo vedendo nel browser.

La risposta a queste esigenze dell'industria, pur se non priva di aspetti positivi e anche (in teoria) tecnicamente efficace, presenta lati discutibili da non sottovalutare, anche per chi non è un sostenitore sfegatato di Open Source e formati aperti. Per comprenderli, occorre fare un passo indietro nella storia dei browser.

Fin dalla loro prima apparizione a metà degli anni '90 si è cercato di usare questi programmi anche come radio e televisori tradizionali, cioè come semplici *ricevitori*, per quanto interattivi, di contenuti distribuiti via Internet. Fino a oggi o quasi, però, le varie versioni di questi servizi sono state altrettante repliche della stessa idea iniziale: aggiungere ai browser plugin su plugin, cioè librerie di terze parti, come RealAudio nel '95 oppure, in tempi più recenti, SilverLight o appunto Flash.

Ognuna limitata a un certo formato e spesso non disponibile per tutti i browser o i sistemi operativi. Il risultato è quello che ormai conosciamo tutti, a volte grave, a volte

## VIDEO STREAMING ONLINE, IN PROPRIO

Condivisione o distribuzione indipendente di video *in streaming*, su Internet e non, non sono più riservate solo a chi può permettersi equipaggiamenti costosi, che magari saranno obsoleti dopo due anni, o a case cinematografiche e grandi reti Tv. Oggi anche una famiglia o una piccola azienda possono accedere a tutti i propri filmati, all'interno di casa, o ufficio, o da Internet. Se poi si utilizzano software Open Source, è possibile farlo in maniera più privata e molto più flessibile di quanto non sia realizzabile con media center commerciali o su portali come YouTube. Vi segnaliamo alcuni prodotti software e metodi generali per svolgere questi compiti.

La prima cosa da tener presente è la differenza stessa fra *streaming*, cioè trasmissione, e mera condivisione di file video. Tecnicamente nulla vieta di scaricare un *intero* video sul proprio computer, come fosse una foto o un file di testo, e iniziare soltanto dopo a vederlo, magari a distanza di ore. In pratica però questo approccio, oltre a essere inapplicabile per video in diretta, è molto inefficiente. Anche avendo connessioni adeguate per scaricare tutto in tempi accettabili, infatti, sarebbe inutile farlo per filmati che abbiamo intenzione di conservare o di cui magari ci interessano soltanto una o due scene. In scenari del genere, più che funzioni di file sharing, servono quelle di un lettore Dvd per trasmettere, un po' alla volta, solo gli spezzoni effettivamente richiesti.

Anche limitandosi allo streaming vero e proprio, abbiamo due situazioni ben diverse da considerare (anche se diversi software di questo tipo ormai le supportano entrambi): trasmettere video solo a Smart Tv e altri dispositivi del genere, che usano direttamente gli standard UPnP (*Universal Plug and Play*, [www.upnp.org](http://www.upnp.org)) e Dlna (*Digital Living Network Appliance*, [www.dlna.org](http://www.dlna.org)) è un conto. Servire su rete domestica, o via Internet, computer, smartphone e tablet generici che invece mostrano i video tramite browser o altri client del genere è un altro. Vediamo alcuni software Open Source per quest'ultimo scenario.

### VLC, IL PLAYER TUTTOFARE

Molti utenti Linux, Mac e Windows, sanno già che VLC Media Player ([www.videolan.org/vlc](http://www.videolan.org/vlc)) può riprodurre file video in parecchi formati, attraverso un'interfaccia grafica semplicissima, per quanto riguarda le funzioni di base. Pochi di loro sanno però che, nascosti nei vari menu e schede di configurazione di questo player, ci sono anche i controlli che servono per trasformarlo in un server per streaming domestico. Se si ha una scheda di acquisizione Tv supportata, si possono ad esempio ritrasmettere a tutti i computer di casa i programmi del canale selezionato. Ancora



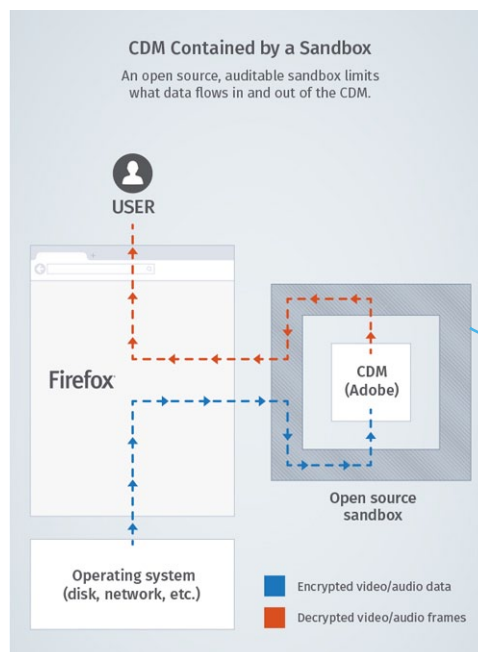
> più facile è trasmettere on demand qualsiasi video conservato su disco locale. Il fatto che il programma sia disponibile anche in versione "Portable", capace di girare da chiavetta Usb, permette infine di mandare in streaming i video sulla stessa chiavetta su *qualsiasi* rete a cui la si connetta. Si noti che queste funzioni rendono Vlc utile anche in scenari didattici, per condividere facilmente documentari, tutorial o altro materiale video con tutti i computer di un'aula di informatica.

I flussi streaming trasmessi da Vlc sono immediatamente utilizzabili, scegliendo il protocollo Mms e il profilo Wmv, anche da client Windows. Se tutti i client sono Vlc, si possono anche adottare configurazioni più sofisticate, per ottimizzare la banda necessaria e migliorare le prestazioni. Impostando, ad esempio, il ritardo programmabile fra le riproduzioni si riduce grandemente l'eco altrimenti causato da diversi computer che riprodurrebbero lo stesso audio, con ritardi diversi!

## STREAM VIDEO DA LINUX SU INTERNET

Il metodo precedente, basato su Vlc, è semplicissimo da usare perché non ha alcun bisogno di un server vero e proprio: basta avviare, quando serve, una copia di Vlc che inoltri il video che tratta ad altri client dello stesso tipo. Per fare vero e proprio streaming però, magari da un vecchio Pc riciclato in casa o in ufficio al proprio smartphone, conviene usare altri strumenti: il tutorial menzionato nel box Risorse spiega in dettaglio come allestire un servizio del genere con Ubuntu e client Android. Il metodo di base, però è assolutamente generico e abbastanza semplice, almeno a livello concettuale.

Il nucleo del sistema è un server Linux su cui giri un qualsiasi Media Center Open Source: questo è il programma che manderà in streaming, a qualunque client ne faccia richiesta, i video messi a sua disposizione in apposite cartelle sul disco. Sul Pc Linux deve girare anche un server Ssh, per comunicare con i client attraverso connessioni cifrate. Nel modem che collega quel server a Internet va poi abilitata la funzione di *port forwarding* alla porta Ssh, che altrimenti sarebbe (giustamente!) invisibile dall'esterno. Fatto tutto questo, cosa meno complicata di quanto potrebbe sembrare, richiedere al server video on demand dal proprio smartphone o tablet è molto semplice. A patto di avere una connessione sufficientemente veloce, ovviamente.



Direttamente dal sito di Mozilla, un diagramma che mostra come contenere i possibili danni risultanti dall'uso forzato di Cdm: farlo girare dentro una sorta di campana di vetro, che limiti fortemente i contatti con il resto del computer.

imbarazzante: siti che "non funzionano più" al primo aggiornamento di software o sistema operativo, necessità di N plugin diversi (se non incompatibili) per la stessa cosa, cioè vedere filmati via Internet, e soprattutto, **molte** più falle di sicurezza di quanto sarebbe stato ragionevole. Uno stato di cose accettabile quando i computer si usavano solo per qualche email o guardare video di gattini, non oggi che sono indispensabili *anche* per lavorare o fare bonifici online.

Negli ultimi mesi Google, Apple, Microsoft e alla fine anche il World Wide Web Consortium (W3C) e Mozilla hanno accettato di risolvere il problema dei plugin alla radice, con le cosiddette *Encrypted Media Extensions (Eme)*, <http://www.w3.org/TR/encrypted-media/>). Queste "estensioni multimediali cifrate" portano i controlli Drm video nel cuore stesso di tutti i browser e dello stesso linguaggio Html, in un modo ormai ratificato dallo stesso W3C.

Lo scopo ufficiale di Eme è "fornire interfacce che controllino la riproduzione di contenuti protetti". In estrema sintesi, questo significa che tutti i contenuti inseriti nei tag "<video>" di Html5 potranno essere visibili *soltanto* tramite UN unico nuovo componente software, già presente e *profondamente* integrato nel browser, che autorizzerà la riproduzione e nient'altro.

Non c'è dubbio che per gli utenti finali questo possa essere un vantaggio rispetto al passato: un browser del genere dà finalmente la certezza, su qualsiasi sistema operativo, di vedere

qualsiasi cosa senza scaricare plugin in continuazione, se i titolari dei diritti lo permettono. Questo però crea nuovi problemi, uno immediato, l'altro di fondo.

Il problema è che il nuovo "super-plugin" (che chiamiamo così solo per semplicità) capace di sbloccare i contenuti Eme non sarà né opzionale, né aperto. Lo standard Eme di W3C, infatti, è open ma specifica *soltanto* come un browser dovrebbe comunicare, tramite librerie JavaScript, con il cosiddetto *Content Decryption Module (Cdm)*. Quest'ultimo componente (prodotto, ma è quasi irrilevante, da Adobe), che sarebbe l'*unico* in grado di decifrare i flussi video scaricati da Internet, è proprietario e privo di documentazioni complete liberamente accessibili.

Nel Web futuro il Cdm si dovrà usare per forza, per fruire dei video da moltissimi siti, e questo potrebbe essere il meno. Quello che conta davvero è che si dovrà avere per forza sul proprio computer, se si vorrà usare un browser moderno in *qualsiasi* modo, un software chiuso, cosa mai successa prima. Ovvero software la cui "sicurezza" si basa soprattutto sull'oscurità dei sorgenti, un'idea dimostratasi fallace innumerevoli volte negli ultimi decenni. Peggio ancora, se qualcuno scoprisse un baco in quel codice e ne segnalasse l'esistenza, ai sensi delle leggi attuali su copyright e informatica come il Dmca, commetterebbe un reato. A quanto risulta dai suoi stessi comunicati ufficiali, la fondazione Mozilla,





Questo esempio dal sito di Daala mostra come i suoi algoritmi di compressione (a destra) possano degradare un'immagine molto meno di quelli dei codec tradizionali (al centro).

che sviluppa Firefox, ha accettato di includere Eme soltanto perché costretto dagli eventi. L'organizzazione teme, ed è difficile darle torto, che i suoi utenti abbandonerebbero in massa Firefox se quel browser perdesse la capacità di accedere a YouTube e a tutti, o quasi, i portali Web dei grandi media. Mozilla ha comunque cercato di minimizzare i rischi creando in Firefox una "area protetta" (*sandbox* in Inglese) in cui rinchiudere il Cdm che si è impegnata ad accettare. Questa *sandbox* permetterà solo alcune comunicazioni con il Cdm, e il suo codice è Open Source, quindi ispezionabile a fondo da esperti indipendenti. Di conseguenza (vedi la figura \$\$\$) il Cdm dentro Firefox avrà meno possibilità di accedere a file sul computer, o di controllarne le funzioni, di quante ne avrebbe su Chrome, Safari o Internet Explorer.

Il problema di fondo a cui avevamo accennato rimane comunque: standardizzare con Eme l'uso dei Cdm significa accettare il principio che poche aziende private (in questo caso le majors cinematografiche e le reti televisive) possono imporre direttamente come un browser deve comportarsi, o cosa deve impedire ai suoi utenti, per essere dichiarato standard. L'organismo indipendente e non a scopo di lucro che finora ha avuto questa responsabilità, il W3C, è costretto ad assumere un ruolo subordinato. Per capire che questo non è un problema

irrilevante per gli utenti finali, né "ristretto" ai filmati, basta considerare che l'architettura di Eme e Cdm è affatto limitata al video. Nulla impedisce, partendo da quella filosofia, di imporre browser che si rifiutino anche di mostrare immagini statiche, o addirittura certi *font*, se i loro *creatori* (prima ancora degli utenti finali) non accettano le condizioni imposte da chi ha sviluppato i Cdm.

### NEL FRATTEMPO, IN TV...

Le ultime decisioni in materia di video via Web non sono le migliori possibili né per gli utenti, né per i produttori indipendenti, sia di video, sia di software innovativo. In salotto, cioè sul fronte del video digitale da TV "precaricato" su dischi o altri supporti fisici anziché ricevuto in streaming, le cose non vanno molto meglio. Come il video online, anche quello "fisso" è basato su *codec* (*coder/decoder*), ovvero coppie di algoritmi software che rispettivamente comprimono il video digitale per farlo entrare sui supporti fisici, e lo decomprimono prima di inviarlo da questi ultimi allo schermo. Un buon codec deve fornire il miglior compromesso possibile fra compressione, qualità dell'immagine decodificata, rapidità di compressione e rapidità di decodifica.

I file video, per esempio quelli con estensione .mp4, sono in realtà semplici *contenitori* di altri file, o flussi, indipendenti: video, audio e sottotitoli in diverse lingue, menu e così via. L'indipendenza dà una flessibilità indispensabile: il formato video migliore per un documentario potrebbe non essere ideale per un cartone animato, e viceversa; ma è anche il motivo per cui, i file (cioè contenitori) potrebbero non funzionare passando da una distribuzione Linux all'altra.

### Il codec si aggiorna

Gli algoritmi di compressione devono fornire il miglior compromesso tra qualità e rapidità

Molti codec e prodotti consumer attuali sono basati su qualche variante o combinazione di quelli chiamati H.264 e Mpeg. Quasi tutti i loro utenti finali già sanno che per loro questi codec sono sostanzialmente freeware, cioè software gratuito ma proprietario.

Meno noto è il fatto che chi produce video con questi codec (per esempio chi riprende un filmato durante proprie vacanze), è sottoposto a clausole molto più stringenti. L'uso di codec Mpeg2 o H.264 è gratuito per i produttori soltanto se il video finale (anche se distribuito in *altri* formati) è disponibile gratuitamente. Inizialmente questa esenzione era valida soltanto fino al 31 dicembre 2015, poi è stata estesa a tempo indeterminato. Nel frattempo arrivava il Blu-Ray,



## RISORSE

Una introduzione non troppo tecnica all'architettura Eme si trova su [www.html5rocks.com/en/tutorials/eme/basics/](http://www.html5rocks.com/en/tutorials/eme/basics/) o in italiano su <http://hi-tech.leonardo.it/drm-in-html5-il-w3c-pubblica-la-bozza-di-encrypted-media-extensions-per-i-video/>. Mozilla ha spiegato la sua strategia per minimizzare i rischi del Cdm nella pagina <https://hacks.mozilla.org/2014/05/reconciling-mozillas-mission-and-w3c-eme/>. Le riflessioni da cui è nato il progetto Lib-Ray sono state pubblicate nell'articolo "Five ideas for escaping the Blu-Ray blues" (<http://fsmsh.com/3511>). L'elenco aggiornato dei browser e delle librerie utilizzabili su Linux per produrre o vedere filmati WebM si trova su [www.webmproject.org/tools/](http://webmproject.org/tools/). La procedura per trasformare un computer con Ubuntu in un server per streaming privato su Android, comprensibile anche se non si conosce bene l'inglese, è descritta in dettaglio nella pagina <http://thepcspy.com/read/remote-streaming-access-xbmc-kodi>.



**Lib-Ray potrà essere distribuito liberamente su schede flash o su qualsiasi altro supporto digitale, file compresi.**

eccellente dal punto di vista della qualità dell'immagine, ma inquietante e oppressivo secondo alcuni, fra cui il programmatore e produttore Terry Hancock.

Chi vuole masterizzare dischi Blu-Ray professionalmente, diceva Hancock nel 2011, deve accettare per forza certe gravi condizioni, indipendentemente dal prezzo finale di vendita dei dischi stessi e dal copyright dei video masterizzati: si è obbligati a inserire *per forza*, su ogni singolo disco, software anticopia che, oltre a costare troppo per piccoli produttori indipendenti, sarebbe inutile e intrusivo.

Hancock, e moltissimi altri con lui, definiscono inutile questo software e i relativi algoritmi (il cui nome, per la cronaca, è Aacs), sia perché sono stati già violati (ed è solo per questo che è possibile, anche se non sempre immediato, vedere film Blu-Ray su Linux), sia perché **tutti** coloro che masterizzano Blu-Ray dovrebbero inserirlo per forza nei dischi (pagandone la licenza): compreso chi, come lo stesso Hancock, per qualsiasi motivo sarebbe ben felice che i suoi film venissero copiati. L'intrusività consiste invece nel fatto che il software Aacs può "attaccare" i lettori Blu-Ray degli utenti, nel senso che potrebbe scaricare contro la loro volontà, o a loro insaputa, tutti i suoi aggiornamenti, anche se inutili o pericolosi.

**LE SOLUZIONI: WEBM, VP8/9, DAALA**

La richiesta di ricevere via Web, almeno in condizioni ottimali, video di qualità paragonabile a quello delle trasmissioni televisive più moderne

da un lato, e quella di alternative ai vincoli appena descritti dall'altro ha prodotto vari risultati, fra cui il primo posto per potenzialità spetta senz'altro allo standard WebM ([www.webmproject.org](http://www.webmproject.org)): un container libero da brevetti e royalty, pensato specificamente per fornire definizioni (relativamente) alte, anche quando viene scaricato e riprodotto con computer limitati e/o da connessioni a Internet lente o instabili. Prestazioni a parte, un qualsiasi video WebM dovrebbe essere immediatamente visualizzabile (Drm permettendo, ovviamente) con qualsiasi browser recente compatibile con Html5. All'interno di un file .webm, infatti, potranno esserci soltanto audio Vorbis e video codificato secondo lo standard VP8 o VP9, che vedremo tra poco. Questa scelta dovrebbe proteggere gli utenti dalle sorprese fin troppo frequenti in altri formati video, cioè l'uso di codec audio/video poco diffusi e supportati.

Il primo codec video di WebM avrebbe dovuto essere quello chiamato VP8, sviluppato dallo stesso gruppo. La mancanza di supporto iniziale e il desiderio di raggiungere risoluzioni più alte han fatto sì che i primi usi di massa di WebM saranno probabilmente già basati sulla versione successiva dello stesso codec: VP9 ([www.webmproject.org/vp9](http://www.webmproject.org/vp9)), presentato l'anno scorso, che

dovrebbe essere in grado di arrivare fino a risoluzioni 4K e che gode dell'appoggio ufficiale di partner come Samsung e Toshiba.

Da parte sua, la fondazione Mozilla ha annunciato che Firefox sarà compatibile con VP9. Allo stesso tempo, anche per non legarsi troppo ad alcuna specifica azienda, ha iniziato lo sviluppo di un codec video alternativo a VP9 e completamente nuovo: Daala (<http://xiph.org/daala>), che non sarà limitato da brevetti poiché utilizza tecniche diverse da tutte quelle presenti nei codec già affermati da decenni, a partire dagli algoritmi matematici di base.

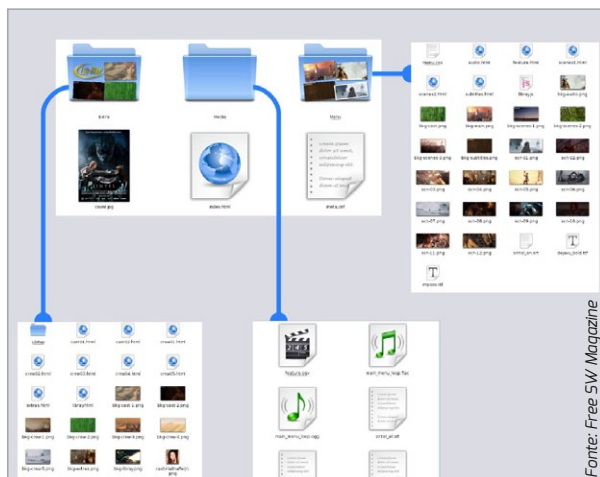
**LIB-RAY: PIÙ APERTO**

Sul fronte che abbiamo definito "da salotto", Hancock ha iniziato a lavorare nel 2011 su un nuovo formato, chiamato Lib-Ray (*Lib...erating, High Definition Video*, <http://lib-ray.org/>), proprio per distinguerlo dalle limitazioni del Blu-Ray. La prima è nel supporto: Lib-Ray non sarà associato a nessun disco o altro specifico oggetto fisico, ma potrà essere distribuito direttamente su schede flash o su qualsiasi altro supporto digitale, file compresi.

Internamente, questo contenitore utilizzerà codec Flac o Vorbis per l'audio, VP8 o VP9 per il video, codice Html5 per il sistema di menu e altri formati ugualmente aperti per sottotitoli e contenuti speciali. La risoluzione di partenza dovrebbe essere Full Hd, espandibile nel futuro fino a 4K (4096x2048). Nessun supporto, ovviamente, per Drm, codici regionali e altre "limitazioni". Allo stesso tempo, e partendo dagli stessi principi, Lib-Ray specifica chiaramente anche come includere metadati utilissimi, almeno per bibliotecari e cinefili, ma apparentemente sgraditi alle case cinematografiche: quelli per l'indicizzazione completa e automatica dei video, anche se copiati su altri supporti fisici. Nelle intenzioni di Hancock, il suo progetto dovrebbe portare al rilascio di un'interfaccia grafica e un player software per creare e riprodurre questi video almeno sotto Linux, una distribuzione preconfigurata come home theater Lib-Ray e un manuale completo per sviluppatori che vogliano creare altri programmi dello stesso tipo.

Al momento, il progetto sembra fermo, ma anche se non dovesse ripartire ha comunque già gettato solide basi per proseguire nello sviluppo di standard video aperti.

Un filmato Lib-Ray sarà molto simile, internamente, a un qualsiasi, normalissimo sito Web: una cartella per il video vero e proprio, una per i menu scritti in Html5 e un'altra per metadati e contenuti extra



# LINUX NEWS



## OWNCLOUD 8, LA NUVOLE PERSONALE SEMPRE PIÙ COMPLETA

**O**wnCloud è un progetto di grande interesse per il futuro del Web. Tecnicamente, è solo un'altra applicazione Lamp, come i blog WordPress: un sito Web dinamico che gira su (L)inux e web server (A)pache, combinando un database (M)ySql e del codice (P)hp. È la funzione offerta che è molto diversa. ownCloud e i suoi plugin creano infatti una versione *integrata e privata* degli stessi servizi base offerti da portali come Dropbox, Google Docs, Google Calendar e Flickr: uno spazio, utilizzabile da qualsiasi computer o smartphone, in cui scrivere testi in formato OpenDocument (lo stesso di Libre Office) conservare file, condividerli con altre persone e in generale lavorare via Internet.

La versione 8 di ownCloud, arrivata a febbraio 2015, contiene parecchie novità, da funzioni di ricerca più efficienti a un miglior

supporto della *federazione*, che è la possibilità di collegare copie *diverse* di ownCloud per condividere facilmente calendari o cartelle di documenti. Vari plugin permettono di sincronizzare un spazio ownCloud con account su Dropbox o Google Document. Per avere un'idea generale di cosa ownCloud può fare e che aspetto ha, è possibile collegarsi al demo online <https://demo.owncloud.org>. Per capire meglio l'importanza della federazione suggeriamo invece di visitare la home page di OpenCloudMesh (<https://owncloud.com/lp/opencloudmesh/>): una rete di "nuvole" private, appartenenti a varie Università di tre continenti, ognuna collegata a tutte le altre proprio con ownCloud. Grazie a OpenCloudMesh, qualsiasi ricercatore con account ownCloud in una qualsiasi università può vedere, senza nessuno sforzo, tutti i file di tutti i suoi colleghi in un unico insieme di cartelle nel browser.

## DEBUG ANDROID DA FACEBOOK

Il suo nome ufficiale è Stetho (come stetoscopio, in inglese) ed è uno strumento presentato da Facebook per effettuare test e controlli in fase di sviluppo di applicazioni per Android. Non si discosta, nell'aspetto, dai tool per lo sviluppo web che abbiamo già visto all'interno di Chrome (cosa apprezzata da chi già li utilizza). La descrizione delle funzionalità è sul sito Facebook (<http://bit.ly/1Bc5QDP>), Stetho è disponibile su Github: <https://github.com/facebook/stetho>.

## UNA LANTERNA MAGICA PER RAGGIUNGERE ANCHE I VIDEO "NASCOSTI" DA FLASH



**L'**applicazione "Linterna Mágica" (<http://linternamagica.org>) è un programma piccolo piccolo, ma che potrebbe facilitare parecchio la vita di molti navigatori del Web. Girando all'interno del browser Firefox, la Linterna rimpiazza automaticamente tutti i link a clip video "nascosti" all'interno di codice Flash con altrettanti link agli stessi filmati, in formato Html standard. In questo modo, ovviamente se il browser ha i codec necessari, è possibile accedere a quei video anche senza attivare il plugin Flash, che in generale rallenta la navigazione e può anche creare problemi di sicurezza.