



Un nuovo OS e un nuovo shop: Firefox

C'è un nuovo sistema operativo libero per telefoni cellulari: Firefox OS. È basato su Linux, come Android, e ha un'interfaccia utente Html5. C'è anche un nuovo canale di vendita per sviluppatori: Firefox Marketplace.

Mozilla Foundation ha presentato, al Mobile World Congress di Barcellona, un sistema operativo per telefoni cellulari, chiamato Firefox OS. Sarà completamente open source, basato su Linux e il browser di Mozilla Foundation. La caratteristica saliente è che l'intera l'interfaccia utente è un'applicazione Html5. Mozilla vanta il supporto di 17 operatori mondiali, fra i quali Telecom Italia Group, Telefónica e Deutsche Telekom, mentre fra i produttori interessati a dotare i propri telefoni del nuovo sistema operativo troviamo Alcatel, LG, ZTE e Huawei. Tutti i dispositivi in sviluppo avranno una Cpu Qualcomm Snapdragon, le stesse di tutti i telefoni Windows e della maggioranza dei dispositivi Android.

Insieme al sistema operativo, prende il via un negozio di applicazioni online, Firefox Marketplace, che promette agli sviluppatori una grande flessibilità, con la possibilità di ospitare le applicazioni su un proprio sito e configurare i sistemi di pagamento, grazie a un protocollo di pagamento aperto e documentato. Firefox Marketplace, infine, può coesistere con gli shop nativi degli altri sistemi operativi. Completa il quadro l'annuncio di una nuova versione

mobile di Firefox, per il sistema operativo Android.

Il sistema operativo

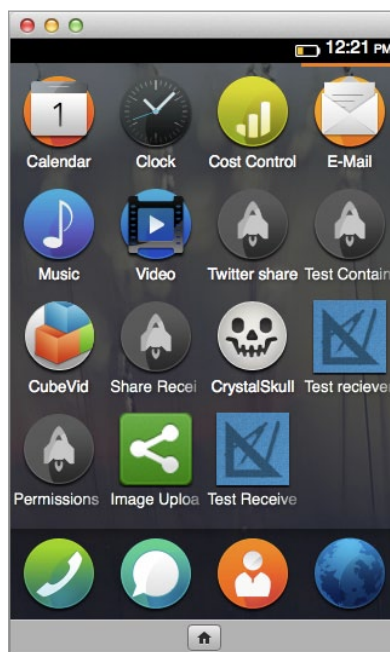
Firefox OS è in gestazione da diverso tempo con il nome di Boot to Gecko, di cui rimangono tracce nella documentazione e nei commenti ove compare spesso la sigla B2G.

Gecko è il nome del motore di rendering delle pagine, che sta al cuore di Firefox. Il nome del progetto nasce dall'idea

di creare una distribuzione Linux con il minimo necessario per arrivare ad avviare un browser dal bootstrap del sistema. In un certo senso si tratta dell'ennesima reincarnazione di un'idea degli anni '90: il Network Computer, che fu a suo tempo al centro delle browser wars e dello scontro frontale fra Microsoft e Netscape.

Il primo contatto con Firefox OS non crea sorprese, l'interazione è piuttosto familiare e intuitiva. Il corredo di applicazioni nel simulatore non è risicato: sono presenti le tipiche applicazioni di corredo di uno smartphone, come il calendario, i contatti e un client di posta e qualche accessorio, come le mappe Here di Nokia e un semplice gioco.

Dal punto di vista dell'architettura, Firefox OS si basa su una versione leggermente modificata di Linux, che include le estensioni al sistema create



Ecco un esempio di interazione con il telefono, ripreso nel simulatore. Le icone sono ancora provvisorie. Il corredo di applicazioni non è scarso.

dal team di Android per supportare i telefoni cellulari, che non fanno parte della distribuzione normale di Linux. Per il resto, il sistema operativo viene definito sostanzialmente di serie nella documentazione di sviluppo.

Sopra Linux troviamo uno strato di software, chiamato Gonk, che offre al browser un ambiente operativo adatto. Gonk, nelle parole della documentazione di Firefox OS, può essere visto come una distribuzione Linux molto semplificata.

In Gonk troviamo alcune librerie standard di Linux, come libusb e bluez e strati di software provenienti da Android, come il supporto per la macchina fotografica e il Gps. Il browser Firefox gira al di sopra di Gonk, che è a tutti gli effetti un porting target per Firefox, come Windows, OS X o Android, come si vede navigando per i sorgenti del livello di astrazione hardware di Mozilla, su github.com/cgjonas/mozilla-central/blob/master/hal.

All'interno del browser gira Gaia, l'interfaccia utente di Firefox OS, che con le sue icone tondeggianti e un gruppo di quattro icone fisse nella barra di lancio principale non si discosta significativamente dall'aspetto di altri sistemi operativi mobili.

La versione disponibile adesso di Gaia è una pre-release e le icone non sono definitive. L'interfaccia utente di Firefox OS può essere esplorata con un simulatore che gira all'interno del browser

Firefox. L'installazione è semplicissima, dato che si tratta di una estensione del browser, disponibile nel menu Strumenti, Componenti Aggiuntivi. È richiesta la versione 19 di Firefox.

Il simulatore è il modo più facile di sperimentare con Firefox OS, anche se supporta solo una parte delle interfacce applicative di Firefox OS, la gestione dei contatti. Per semplici applicazioni che interfacciano servizi web è sufficiente. Per lo sviluppo di applicazioni più vicine all'hardware ci sono due emulatori a disposizione, uno dei quali emula la Cpu ARM di riferimento, mentre l'altro gira nativamente su processore Intel. Entrambi richiedono di installare l'intera catena di sviluppo su un sistema Linux. Infine, vi è un simulatore per Windows, b2g.exe, che però non siamo riusciti a eseguire su un sistema Windows 8.

Le istruzioni per gli sviluppatori mostrano come installare una immagine di Firefox OS su uno dei terminali di riferimento, per esempio il Galaxy S2 o il Nexus S di Samsung. Per installare Firefox OS la prima volta, occorre un device con installato Android 4.

Le interfacce applicative

La maggior parte delle funzionalità applicative sono quelle di HTML5, con il supporto per il disegno della *canvas* e le funzioni grafiche accelerate di WebGL a cui si aggiungono le trasformazioni e le

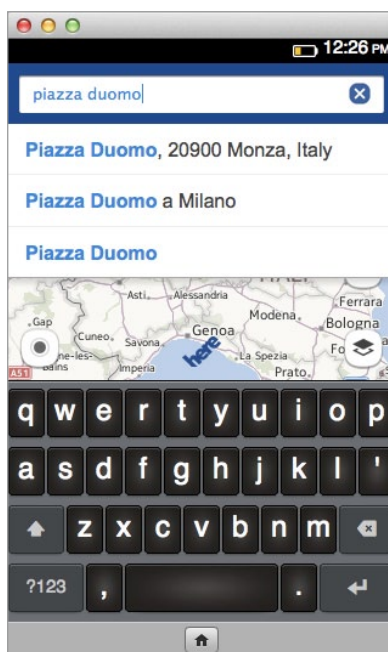
animazioni CSS e il supporto per i tag *audio* e *video* di HTML5.

Tutto questo è parecchio, ma naturalmente non basta, per questo Gonk aggiunge uno strato di supporto, corredato dai binding di linguaggio per Javascript, per le funzioni specifiche di un telefono, per esempio la libreria Audio Policy consente di gestire il canale audio e disciplinare l'accesso all'audio di lettori musicali, l'audio proveniente da una conversazione telefonica e gli allarmi.

La memorizzazione locale di audio, video, immagini e filmati è disciplinata dalla libreria Device Storage e, naturalmente, ci sono funzioni specifiche per la gestione dei contatti e della geolocalizzazione. Infine, lo strato di networking è stato potenziato rispetto al rudimentale supporto che offre il browser, come è necessario per un sistema operativo. In ultimo, abbiamo le interfacce applicative per interagire con il Marketplace, un canale cruciale per il successo del sistema. Rimane un punto fermo, su cui si fondano le prospettive di successo del nuovo ambiente: un programmatore web che non sia rimasto fermo all'impaginazione con le tabelle conosce il 90% di quello che serve per sviluppare applicazioni in Firefox OS.

Il Marketplace

L'altra novità interessante è l'apertura di un negozio online, visibile all'indirizzo marketplace.firefox.com, destinato a



Firefox OS, ma anche alle varie versioni mobili del browser Firefox. Il negozio è aperto a piattaforme e device diversi, ma richiede il browser Firefox.

Come avviene in altri shop, per esempio l'AppStore di Apple, le applicazioni acquistate possono seguire l'utente su diversi dispositivi e sistemi operativi.

Il minimo richiesto per la pubblicazione di un'applicazione è la creazione di un manifest, in formato json, che descrive l'applicazione e deve trovarsi allo stesso livello della pagina di ingresso. Ecco un esempio di manifest:

```
{
  "name": "My App",
  "description": "Una breve
descrizione",
  "launch_path": "/",
  "icons": {
    "128": "/img/icon-128.png"
  },
  "developer": {
    "name": "Il mio nome e
l'organizzazione",
    "url": "http://la-mia-homepage.
org"
  },
  "default_locale": "it"
}
```

Il manifest consente di specificare i linguaggi a disposizione e di elencare i diritti che l'applicazione richiede, per esempio la pubblicazione di notifiche sul desktop, l'accesso alla posizione geografica o alla rubrica.

Si possono creare applicazioni a diverso livello di privilegio: web, privileged e certified. Al livello più basso, il codice gira con il livello di privilegio che ha un'applicazione nel browser, quindi non è in grado di fare danni all'utente e può essere messo in linea senza particolari controlli, fatti salvi determinati controlli formali, per esempio la validazione del manifest e la identica provenienza del manifest e del codice dell'applicazione.

Ai livelli superiori occorre la validazione del codice da parte del Marketplace per verificare il rispetto della sicurezza, come avviene nel negozio di Apple.

Un'applicazione marcata come certified deve essere approvata dal costruttore del device o dal carrier. Questo privilegio è riservato a applicazioni che offrono funzionalità di base del telefono. L'installazione di un'applicazione è molto semplice: ecco un ritaglio di

codice che verifica che un'app non sia già installata

```
var request = navigator.mozApps.
getSelf();
request.onsuccess = function() {
  if (request.result) {
    // già installata
  } else {
    // non installata
  }
};
request.onerror = function() {
  alert('Error checking
installation status: ' + this.
error.message);
};
```

l'installazione vera e propria richiede pochissimo codice:

```
var request = navigator.mozApps.
install("http://path.to/my/example.
webapp");
request.onsuccess = function() {
  // installazione corretta
};
request.onerror = function() {
  // errore. controllare this.
error.name
};
```

Per lo sviluppatore è semplice controllare che l'utente abbia il diritto di eseguire l'applicazione, per esempio con questo codice nell'avvio dell'applicazione:

```
mozmarket.receipts.Prompter({
  storeURL: "https://marketplace.
firefox.com/app/your-app",
  supportHTML: 'email you@yourapp.
com',
  verify: true });
```

L'interfaccia di pagamento è aperta e documentata su wiki.mozilla.org/WebAPI/WebPayment.

«Realizzare applicazioni mobili con un nucleo Html5 permette di coprire tutti i sistemi operativi da iOS a Windows RT»

Cosa ci sembra al primo contatto

L'idea di un intero ecosistema di applicazioni scritto in Javascript e in esecuzione in un browser sembra presa da uno dei più fantasiosi futurismi degli anni '90, ma in realtà non è pellegrina, ora che le Cpu come Snapdragon sono così potenti, gli interpreti Javascript così veloci e l'ambiente operativo del browser così ricco.

Il multitasking è stato allegramente cestinato in tutti i sistemi operativi mobili, che dedicano solo un minimo di cicli di clock alle applicazioni che non hanno una finestra in primo piano, se proprio ne dedicano, anche la memoria delle applicazioni in background viene allegramente riciclata quando c'è scarsità. Firefox OS non dovrebbe quindi scontare la sua peculiare architettura in termini di prestazioni e velocità di risposta. Naturalmente, sospendiamo il giudizio fino al momento di vedere una release definitiva all'opera su un telefono fisico. Dal punto di vista degli utenti, ci aspettiamo una rapida proliferazione di terminali, perché Firefox OS potrebbe essere meno costoso di Android, sul quale gravano dei diritti di autore di Microsoft che sono stimati intorno a quindici dollari per copia. Se Firefox OS riesce a dribblare eventuali cause per diritti di autore, potrebbe avere il doppio vantaggio di non portarsi dietro gli interessi di Google e i diritti di licenza di Microsoft, diventando una scelta ideale per i terminali economici e per i mercati emergenti. I primi telefoni con la volpe di fuoco usciranno per l'appunto in Brasile, Colombia, Ungheria, Messico, Montenegro, Polonia, Serbia, Spagna e Venezuela.

Dal punto di vista degli sviluppatori, invece, c'è un altro negozio online, facile da utilizzare, flessibile e basato su standard aperti, da riempire di software. Da un punto di vista strategico, realizzare applicazioni mobili con un nucleo Html5 sembra una possibilità da considerare seriamente, perché è vero da un lato che si perde dal punto di vista delle prestazioni, ma dall'altro lato, si riescono però a coprire tutti i sistemi operativi, da iOS a Windows RT. Inoltre le interfacce Html si possono personalizzare dal telefono al tablet con pochi tocchi di foglio di stile. Insomma, prima di decidere per un'interfaccia nativa bisogna avere un motivo tecnologico misurabile.