

Linux: dal Web all'industria

Dopo aver conquistato una posizione di rilievo nel settore dei server di rete e aver intaccato il mondo dei desktop, Linux si sta rapidamente diffondendo tra le applicazioni dedicate, ambendo anche all'unificazione mediante uno standard comune

MASSIMO GIUSSANI

La tecnologia dell'informazione ha da tempo valicato i confini della mera automazione del lavoro d'ufficio diffondendosi in ambiti che vanno dalla gestione d'impresa all'automazione dei processi produttivi, fino agli stessi prodotti finiti che racchiudono al loro interno, in maniera spesso trasparente all'utente, processori più o meno complessi. I mainframe che gestiscono flussi di produzione e

ze specifiche molto diverse tra loro. Per quanto concerne l'automazione del lavoro, ad esempio, può venire richiesto un funzionamento deterministico in tempo reale; i produttori di dispositivi palmari o di telefoni cellulari vedranno invece con maggior favore un sistema operativo altamente modulare dalle dimensioni contenute e con la possibilità di interfacciarsi all'hardware più recente.

Fino a qualche anno fa la maggior parte dei sistemi operativi embedded e real-time veniva realizzata su misura di applicazione dagli stessi produttori di apparecchiature e solo poche case software (peraltro sconosciute agli 'informatici d'ufficio') potevano vantare una diffusione di una certa entità.

Il sempre più frequente aggiornamento dei componenti hardware e l'esigenza di ridurre il time to market hanno reso sempre più onerosa la realizzazione di sistemi operativi ad hoc e hanno incrementato la richiesta di sistemi operativi cosiddetti 'off the shelf'. Ed è qui che, dopo aver conquistato il mercato dei server e aver intaccato quello dei PC desktop, entra in scena Linux. Il sistema operativo rappresentato dal pinguino ha infatti catturato l'attenzione della maggioranza degli operatori abituati a sviluppare essenzialmente da zero i sistemi operativi per le proprie applicazioni. La flessibilità, la modularità e la robustezza di Linux, associate al continuo aggiornamento dei driver di periferica, eredità della diffusione nel mondo dei server e dei PC, fanno di Linux una base ideale per lo sviluppo di sistemi operativi embedded e, con le dovute accortezze, anche in tempo reale.



catene di approvvigionamento, i microcontrollori che gestiscono le isole di lavorazione e i microprocessori nascosti all'interno dei prodotti dell'elettronica di consumo richiedono sistemi operativi in grado di soddisfare esigenze

Linux e la filosofia 'open-source'

Linux vede la luce nel 1991 in Finlandia per mano di Linus Torvalds, uno studente dell'Università di Helsinki che, poco più che ventenne, decide di scrivere un kernel ispirato a Minix, un sistema operativo didattico simile a Unix. Invece di tenere per sé il proprio lavoro, Torvalds lo condivide con la comunità di programmatori che popolava, e tuttora popola, i gruppi di discussione di Usenet. Le principali differenze tra Minix e Linux risiedevano nella differente architettura di sistema (a microkernel il primo, a kernel monolitico il secondo) e nell'approccio diametralmente opposto riguardo alla commercializzazione: pur essendo un prodotto didattico, Minix non era disponibile gratuitamente. Uno dei motivi del successo di Linux risiede appunto nella modalità con cui Torvalds ha messo a disposizione della comunità di programmatori il proprio lavoro: la licenza GPL del progetto GNU. Per comprendere cosa significhi, facciamo un ulteriore salto indietro nel tempo a quando il sistema operativo Unix, creato presso i laboratori Bell da un gruppo di ricercatori guidati da Brian Kernighan, Dennis Ritchie e Ken Thompson, veniva distribuito ai ricercatori e alle università in cambio di una mera quota simbolica. Con l'avvento della commercializzazione del software, anche Unix divenne un sistema proprietario e, oltre ad essere costoso, veniva distribuito senza il codice sorgente, impedendo di fatto ai programmatori di analizzarlo e di apportarvi modifiche. La leggenda vuole che Richard Stallman, allora ricercatore del MIT, abbia concepito l'idea del progetto GNU (acronimo ricorsivo che sta per Gnu Not Unix) in seguito all'impossibilità di modificare un driver di stampa per l'assenza del codice sorgente. Il progetto GNU (www.gnu.org) nasce nel 1984 con l'ambizioso intento di produrre un sistema operativo della stessa levatura di Unix, che fosse però liberamente accessibile e modificabile da tutti. Al concetto di 'copyright' Stallman contrappose infatti quello di 'copyleft', basato sulla possibilità di copiare, modificare e ridistribuire il software in accordo a una licenza d'uso che estenda la condivisione delle informazioni a tutta la comunità di programmatori. Le modalità con cui le summenzionate operazioni potevano e possono tuttora avvenire è regolamentata da precise licenze d'uso, la più diffusa delle quali è indubbiamente la General Public License (GPL). In linea di massima il software soggetto alla licenza GPL viene fornito con il codice sorgente e chiunque può apportarvi le migliorie o le modifiche che crede, a patto di renderle pubbliche con il relativo codice qualora decida di distribuire a terzi il proprio lavoro. Il programmatore detiene i diritti sul codice che ha scritto ma lo concede in uso alle stesse condizioni di 'libertà di modifica' con le quali ha ricevuto il materiale che ha a sua volta modificato. Questo è quello che viene chiamato software 'open-source' o con codice sorgente libero che, è bene notarlo, non preclude la possibilità di richiedere un pagamento per il proprio lavoro. Tuttavia, e questo può essere elemento di preoccupazione per le aziende che non intendono condividere il proprio know-how, il codice sor-

gente, e pertanto gli algoritmi adottati nel programma, saranno visibili a chiunque, concorrenza compresa. Questi vincoli riguardano ovviamente solo i prodotti che si basano sulle componenti dotate di licenza GPL e che vengono distribuiti al pubblico (o ai clienti): i pacchetti applicativi che si appoggiano a un sistema operativo come Linux sfruttandone solamente i servizi possono però essere realizzati e commercializzati con licenze di tipo tradizionale, distribuendo solo i file binari e occultando il codice sorgente (un esempio è il programma di calcolo Mathematica commercializzato da Wolfram Research). Nel caso di un'applicazione embedded, la tecnologia proprietaria risiede essenzialmente nell'hardware o nel codice applicativo e pertanto l'impiego di Linux non pone delle minacce alla tutela della proprietà intellettuale. La questione delle licenze è tuttavia delicata e deve essere valutata con cura, in quanto è necessario distin-



rete, Linux si mostra in tutta la sua potenza e stabilità al punto da poter vantare la fetta maggiore del mercato, anche a fronte di un costo virtualmente nullo. Sono sempre di più le software house che offrono soluzioni per la gestione aziendale e d'impresa basate su Linux: IBM ed Oracle, per citare due dei nomi più noti, hanno effettuato importanti investimenti in questo campo e le recenti operazioni di marketing hanno saputo portare il pinguino all'attenzione del grande pubblico. Tuttavia, per quanto concerne il mercato dei Personal Computer, è poco probabile che si verifichi l'inversione di tendenza nella quale i concorrenti di Microsoft sperano: il gigante di Redmond rappresenta ancora la scelta predominante per il desktop casalingo e del piccolo ufficio, nonostante Linux stia guadagnando giorno dopo giorno nuovi adepti, anche grazie alla possibilità di installare il prodotto a fianco del sistema operativo della concorrenza. Sensibilmente diversa è invece la situazione nelle aziende, dove la scelta di Linux può essere fortemente motivata dall'abbattimento del costo delle licenze e imposta dall'amministrazione, eventualmente passando per una fase di transizione con due sistemi operativi. Linux può contare su strumenti di produttività per l'ufficio che rappresentano una valida alternativa ai prodotti della suite Office,



guere in che modo il software utilizza o eventualmente altera il codice di Linux. Informazioni più circostanziate possono essere reperite sui siti della comunità 'open-source' (www.opensource.org) e della Free Software Foundation (www.fsf.org).

La diffusione di Linux

Linux si è diffuso a livello mondiale prima di tutto tra i server di rete, specialmente grazie al contributo di Apache, il server Web nato nel 1994 sotto la guida dell'allora ventunenne Brian Behlendorf, un programmatore guidato dagli stessi principi della filosofia 'open-source' che avevano motivato i contributi di Stallman e Torvald. Tra i server di

anche se la scelta di software applicativi di alto livello è ancora abbastanza limitata. Va segnalato, in questo ambito, il tentativo di standardizzare le numerose distribuzioni al fine di perseguire la massima interoperabilità del software: la Linux Standard Base (www.linuxbase.org) è il risultato degli sforzi del Free Standards Group in questa direzione. Ultimamente Linux sta guadagnando parecchie posizioni come sistema operativo per le applicazioni dedicate o embedded. A differenza del mondo dei desktop, in questo settore le previsioni sono più che rosee: una ricerca di mercato effettuata da Venture Development Corporation (VDC) prevede per Linux una posizione di dominio nel panorama dei sistemi operativi embedded, posizione che

sarà raggiunta a discapito della categoria di sistemi operativi proprietari realizzati su misura di applicazione. Ciò che storicamente ha sempre rappresentato la fetta predominante del mercato dei sistemi operativi embedded verrà così ridotta drasticamente e relegata a un ruolo marginale. La recente proposta di uno standard unificato avanzato dall'Embedded Linux Consortium, con l'appoggio di oltre 120 aziende di settore, fornisce un'ulteriore spinta in questa direzione per tutta una serie di prodotti che spaziano dai telefoni cellulari ai sistemi di controllo dedicati. Infine, grazie allo sviluppo della piattaforma .GNU ('dot gnu'), Linux si propone anche come punto di riferimento per lo sviluppo di soluzioni di rete distribuite, in diretta concorrenza con la tecnologia .NET di Microsoft.

Le ragioni del successo

Le proiezioni di Venture Development Corporation stimano che entro il 2005 il 47% circa dei progetti embedded farà uso di Linux, muovendo un fatturato di oltre 300 milioni di dollari. E' ragionevole ipotizzare che il successo pronosticato, finora confermato dalla crescente diffusione, nel settore embedded sia dovuto proprio alla maturità raggiunta da Linux e dalle sue componenti: gli sviluppatori possono infatti realizzare il proprio sistema operativo dedicato partendo da una base di codice ampiamente collaudato da un'intera collettività di programmatori. La disponibilità del codice sorgente, la possibilità di personalizzarlo per soddisfare le esigenze specifiche del proprio progetto, l'assenza di royalty sul kernel e sui principali moduli, nonché la ridotta curva di apprendimento e la facile reperibilità di supporto tecnico qualificato fanno di Linux un temibile concorrente anche delle soluzioni più blasonate. Linux offre, inoltre, di default tutta una serie di strumenti di sviluppo professionali di alto livello, come i compilatori C, C++ e Java, e una documentazione estesa e dettagliata. La possibilità di accedere al codice sorgente e di poterlo modificare in ogni sua parte è particolarmente preziosa nello sviluppo di sistemi dedicati che devono potersi adattare a configurazioni hardware spesso uniche. Altri elementi a favore di Linux sono la sua quasi leggendaria stabilità, la predisposizione all'interfacciamento di rete e il supporto multiprocessore, derivanti dall'ispirazione al modello UNIX. Anche le aziende per le quali la sicurezza del software è di basilare importanza possono trarre vantaggio dall'adozione di Linux: può sembrare paradossale, ma l'apertura del codice permette di verificare gli eventuali punti deboli nei confronti di attacchi da parte di pirati informatici e di utilizzarlo come base per sistemi sicuri. Infine, nonostante Linux non sia proprietà di alcuna azienda, il supporto tecnico può contare su un numero sterminato di sviluppatori in tutto il mondo che rendono disponibili soluzioni ad eventuali problemi (specie se inerenti alla sicurezza del sistema) praticamente in tempo reale attraverso forum di discussione, newsgroup e mailing list. Sono inoltre sempre più numerosi i servizi di supporto tecnico a pagamento che rappresentano a loro volta un'opportunità di guadagno per le aziende del settore.

Le distribuzioni per le applicazioni embedded

Uno dei motivi che ha reso possibile l'adattamento di Linux alle esigenze dei progettisti di sistemi dedicati è la sua intrinseca modularità. Nelle sue diverse implementazioni embedded, Linux può utilizzare solo i moduli strettamente necessari alla particolare applicazione cui il sistema è dedicato. E' possibile, ad esempio, racchiudere le funzioni essenziali del kernel, completo di sistema di gestione della memoria virtuale, a un centinaio di kB, portando a distribuzioni 'complete', pur se minimali, che occupano poco meno di un floppy disk e possono essere caricate direttamente da ROM o da Flash. La varietà di distribuzioni dedicate al mondo embedded è davvero impressionante, sia per l'ampio spettro di applicazioni alle quali sono indirizzate, sia per la copertura di piattaforme hardware che

sono state aggiunte all'architettura Intel x86 sulla quale Linux è stato concepito. Il pinguino di Linux ha saputo migrare su microprocessori PowerPC, Arm, StrongArm, Mips, 68k, SH, nonché su svariati microcontrollori, DSP e System on Chip (SoC). Tutto ciò ha avuto un effetto positivo sugli utilizzatori nel segmento embedded, in quanto ha spianato la strada all'implementazione su un numero crescente di schede dedicate e SBC; è ormai possibile trovare Linux, oltre che sulle piattaforme PCI-ISA, CompactPCI, e PC/104 anche su schede VME e all'interno di dispositivi dedicati al mondo consumer come set-top box, lettori Mp3, Pda, telefoni cellulari e persino orologi. La tabella 1 mostra una succinta selezione delle distribuzioni embedded di tipo open-source e commerciali; un elenco più esaustivo può essere reperito in rete ad esempio su www.linuxlinks.com. Recentemente l'Embedded Linux Consortium (www.embedded-linux.org) ha avviato un processo di standardizzazione delle distribuzioni Linux embedded con l'intento di limitare la frammentazione del mercato, che di fatto ostacola lo sviluppo di soluzioni modulari 'off the shelf' su vasta scala. Lo sviluppo di componenti software compatibili passa necessariamente per il rispetto delle linee guida di uno standard e le specifiche messe in rete da ELC intendono appunto conseguire l'obiettivo di un sistema operativo dal codice liberamente disponibile e in grado di interfacciarsi con i pacchetti applicativi sviluppati da terzi, a prescindere dalla particolare distribuzione adottata. La specifica ELCPS (Embedded Linux Consortium Platform Specification) definisce un'API standard in grado di garantire la portabilità del codice applicativo verso differenti distribuzioni di Linux, garantendo in questo modo la possibilità di utilizzare un determinato prodotto su diversi sistemi con una semplice ricompilazione. I dettagli sulla specifica si possono trovare su www.linuxdevices.com/files/misc/ELCPS_v1.pdf. Lo sforzo di ELC viene supportato dalle numerose aziende che costituiscono il consorzio (tra le quali figurano IBM, Intel, Motorola, Panasonic, Siemens e Sony) e ha visto l'adesione di alcuni dei produttori di importanti distribuzioni embedded o real-time: degna di nota è la recente adesione di FSMLabs che ha deciso di adattare la propria distribuzione RTLinux allo standard. Il successo dell'iniziativa è comunque nelle mani della comunità di programmatori open-source, che è notoriamente molto guardinga nei confronti dei tentativi di assoggettare il codice che produce a restrizioni simili a quelle dei prodotti commerciali.

Le distribuzioni per le applicazioni real-time

Per quanto riguarda le applicazioni dedicate in tempo reale, e in particolare quelle cosiddette 'hard real-time', il discorso delle distribuzioni si fa più complesso, in quanto il kernel di Linux non è stato concepito per soddisfare i vincoli temporali di queste applicazioni. Ricordiamo che un sistema si dice 'soft real-time' se è in grado di portare a termine un dato compito rispettando mediamente i vincoli temporali prefissati, mentre si dice 'hard real-time' se deve

Distribuzioni embedded

Distribuzione	Riferimento
Open-source	
Embedded Debian Project	Debian www.emdebian.org
ETLinux	Prosa www.prosa.it/etlinux
Freesco	Freesco www.freesco.org
Midori Linux	Transmeta http://midori.transmeta.com
muLinux	muLinux http://sunsite.auc.dk/mulinux/
ThinLinux	ThinLinux www.thinlinux.org/
uCLinux	uCLinux www.uclinux.org
Commerciali	
BlueCat	LynuxWorks www.bluecat.com
DSP Linux	RidgeRun www.ridgerun.com
Embedded Linux	Red Hat www.redhat.com/embedded/
Embedix	Metrowerks www.metrowerks.com/MW/Develop/Embedded/Linux/default.htm
Hard Hat Linux	Montavista www.mvista.com
Linux GPL	TimeSys www.timesys.com
RedIce	RedSonic www.redsonic.com

Tab. 1 - Distribuzioni embedded

rispettare tutti i vincoli imposti con una temporizzazione precisa e stabile. Un sistema soft real-time può tollerare un ritardo o un anticipo, a patto che le prestazioni medie si mantengano a un livello accettabile; un sistema hard real-time, invece, potrebbe venire distrutto se uno dei compiti assegnatigli venisse portato a termine troppo presto o troppo tardi. Esistono due possibili approcci per rendere deterministico un sistema operativo che intrinsecamente non lo è: apportare delle modifiche al kernel per aumentarne la risoluzione temporale, oppure utilizzare un modulo dedicato che prenda il controllo del sistema, gestisca gli eventi in tempo reale e utilizzi Linux come un processo preemptibile a bassa priorità. Il primo approccio è esemplificato dalla distribuzione messa a punto dall'Università del Kansas e va sotto il nome di KURT (Kansas University Real Time) Linux: modificando il meccanismo di temporizzazione all'interno del kernel è possibile ottenere risoluzioni temporali dell'ordine di pochi microsecondi. Questo approccio presenta lo svantaggio di fondo di perdere il contatto con il sistema operativo originale e quindi con l'aggiornamento continuo e il supporto che ne deriva. E' comunque un'opzione valida per quelle distribuzioni commerciali come Hard Hat Linux per le quali viene fornito supporto a pagamento (pur fornendo il codice sorgente). Differente è il modo di agire di sistemi come RTAI e RT Linux, il primo messo a punto al Politecnico di Milano e il secondo all'Università del New Mexico. In questo caso si utilizza un secondo kernel che prende il controllo della gestione degli interrupt e favorisce i task in tempo reale rispetto ai processi standard gestiti da Linux. Quando uno dei task in tempo reale deve essere eseguito, Linux abbandona le operazioni in atto e restituisce immediatamente il controllo al kernel in tempo reale che si occupa di soddisfare in maniera deterministica le richieste prioritarie. La tabella 2 riporta una selezione di alcune delle distribuzioni di Linux rivolte alle applicazioni in tempo reale; per un elenco più esteso si può far riferimento al sito www.linuxdevices.com.

Valutare l'impatto economico

Linux non è necessariamente gratuito: alcune distribuzioni presentano infatti dei costi di licenza paragonabili a quelli dei sistemi operativi embedded commerciali. Tuttavia, proprio perché è possibile trovare distribuzioni gratuite con prestazioni simili, il valore aggiunto (rappresentato dal supporto, dai pacchetti applicativi, dalla libreria di driver per dispositivi particolari) deve essere chiaramente percepibile per poter consentire la permanenza sul mercato. In un settore in rapida evoluzione come quello delle applicazioni embedded, la riduzione del time to market può risultare vitale nel decretare il successo di un progetto: la disponibilità di driver di periferica per l'hardware più recente, di strumenti di sviluppo rapido allo stato dell'arte e di un supporto che permetta di risolvere rapidamente i problemi sorti in fase di sviluppo sono oro per i progettisti e fanno storicamente parte del portafoglio di offerte dei produttori di sistemi operativi commerciali. Linux non è comunque da meno: al di là della presenza di una comunità di program-

matori open-source che possono costituire una forma di supporto distribuita sull'intera rete Internet (ma che, ragionevolmente, non si sobbarcheranno i problemi altrui gratuitamente) sono oramai moltissime le aziende che offrono supporto qualificato, ovviamente a pagamento, agli sviluppatori amanti del pinguino. Aumentano anche gli strumenti di sviluppo rapido per velocizzare la realizzazione delle applicazioni e, resta ben solida la base di ottimi compilatori messi a punto dalla comunità di programmatori. Un ulteriore punto a favore del pinguino è rappresentato dal fatto che una piattaforma host completamente basata su Linux e su strumenti di sviluppo open-source può abbattere drasticamente i costi di realizzazione del software (anche dell'80% secondo alcuni analisti) consentendo una politica più aggressiva dal punto di vista dei prezzi; il fatto poi che Linux non richieda il pagamento di royalty rappresenta un ulteriore importante fattore a vantaggio di tutti quei progetti che prevedono l'implementazione su un elevato numero

Distribuzioni real-time

Distribuzione	Riferimento
Open-source	
KURT	Kansas University www.ittc.ku.edu/kurt/
RTAI	DIAPM www.rtai.org
RTLlinux	New Mexico University www.rtlinux.com
Commerciali	
BlueCatRT	LynuxWorks www.bluecat.com
LinuxRT	TymeSys www.timesys.com
Montavista Linux extensions	Montavista www.mvista.com/products/rtp.html
Open RTLlinux	FSMLabs www.rtlinux.com
RedICE Linux	RedSonic www.redsonic.com

Tab. 2 - Distribuzioni real-time

di dispositivi come i palmari, i lettori Mp3 o i telefoni cellulari. La formazione è agevolata dalla diffusione del sistema operativo su server e PC, anche se per le distribuzioni più specifiche può richiedere dei corsi specialistici a pagamento. In ultimo, va osservato che, rispetto ai sistemi operativi commerciali, per la sua natura libera e non soggetta a un controllo centrale, Linux può subire delle modifiche del kernel che possono richiedere l'adeguamento delle applicazioni precedentemente sviluppate, pena la perdita della conformità alle distribuzioni ufficiali e dei vantaggi del supporto diffuso. ■