

BECCARIA, MARCHETTI-STASI
PARRELLA, SOMMA [NMI CLUB]

NOCOPYRIGHT

A cura del *NMI Club*:

Antonella Beccaria (giornalista)

Francesco Marchetti-Stasi (programmatore)

Bernardo Parrella (giornalista)

Emmanuele Somma (programmatore)

Nella programmazione, *NMI (Non Maskable Interrupt)* indica un evento che interrompe l'attività di un microprocessore di fronte a condizioni che hanno priorità su ogni altra attività e che non possono essere ignorate.

NMI Club si propone di approfondire alcuni temi caldi dell'attuale panorama tecnologico, rileggendoli attraverso le inevitabili ricadute socio-culturali ad ampio raggio, con un linguaggio chiaro e diretto per un'informazione alla portata di tutti. Possiamo forse permetterci di ignorare l'attività di *NMI Club*?

Contatti, collaborazioni, proposte: info@nmi-club.org

Versione elettronica del libro: <http://www.nmi-club.org/libri/nosco>

La versione elettronica del volume viene rilasciata sotto la licenza Creative Commons: Attribution-

Commercial-ShareAlike (<http://creativecommons.org/licenses/by-sa/1.0/>).

Prefazione

Questo è un libro sull'informatica, ma non un libro di informatica. È un libro rivolto agli appassionati di informatica, che usino Linux, FreeBSD, MacOS, Windows o qualunque altro sistema operativo; è un libro per gli amanti della tecnologia, ma anche per chi è semplicemente affascinato dalle nuove frontiere della comunicazione. È un libro sulla nascita, la crescita, il destino di un sistema operativo che può fare la differenza, almeno nel settore informatico, tra il capitalismo selvaggio e lo sviluppo sostenibile.

Nello specifico, questo lavoro ripercorre le vicende che hanno portato e fanno da sottofondo al caso "SCO contro Linux," un caso che va scuotendo non solo l'ambito del software libero/open source ma, appunto, anche l'intero mondo informatico. Dove in pratica si tenta di trasformare la banale violazione di un contratto in una guerra termonucleare totale contro Linux, inclusi programmatori, aziende e utenti finali. L'intricata vicenda si dipana attraverso l'analisi di tutti i principali momenti del caso, scavando nel passato recente e meno recente degli anni precedenti l'arrivo di Santa Cruz Operations, e mettendo in luce i diversi elementi in gioco: la storia di Unix con annesse diramazioni, dalla genesi all'esodo fino all'avvento di GNU/Linux; i tortuosi percorsi che hanno portato da Novell a Caldera a SCO, passando per le idiosincrasie di persone quali Ray Noorda e Darl McBride; le specifiche posizioni del querelante e le reazioni della comunità open source/software libero, con stralci da documenti ufficiali e interventi online; una cronologia più che dettagliata affiancata da una

marea di link sparsi, onde documentarsi al meglio e restare sintonizzati.

*Il tutto per offrire al lettore un percorso a fasi alterne (nella definizione del genere *avvocatesco*: *legal story*), non privo di colpi di scena, trovate ad effetto, situazioni al limite del paradossale. E che rimane frutto di un gran lavoro a più mani (e menti), aperto ai contributi futuri e rilasciato sotto la licenza *Creative Commons Attribution-Commercial-ShareAlike*.*

La storia di Unix, e delle battaglie che si sono combattute intorno ad esso negli ultimi trent'anni, potrà non risolvere i problemi più seri dell'umanità nel terzo millennio, ma costituisce un modello di sviluppo che ci auguriamo altri settori possano seguire. Così come i problemi e le politiche che ne hanno segnato le tappe speriamo possano servire da monito per quanto ci aspetta nei decenni a venire, soprattutto ma non solo in campo informatico. Perché in realtà il caso "SCO contro Linux" non fa altro che riflettere la storia esemplare delle distorsioni in tema di 'proprietà intellettuale', dello snaturamento culturale che le grandi multinazionali vanno operando ai danni degli stessi consumatori loro clienti. Da qui, l'urgenza di una rilettura ragionata e motivata di una storia di malaffare nell'odierna società dell'informazione: per non farsi cogliere impreparati di fronte all'attacco generalizzato alla libertà di circolazione delle idee (e del software).

Marzo 2004

Unix, questo sconosciuto

Sebbene le vicende centrali di questo libro siano molto recenti, per comprenderle appieno è necessario ritornare indietro di oltre trent'anni: una scala di tempi che, alla velocità con cui si muove l'informatica, ci porta a un'epoca quasi preistorica. Prima di questo tuffo nel passato, dobbiamo però mettere in prospettiva l'argomento, e cercare di capire, in termini non specialistici, cosa sia un sistema operativo, e cosa sia Unix.

Unix e il resto del mondo

Ancor più di altri termini analoghi che circondano la recente storia dell'informatica, c'è una grande confusione che circonda il termine "sistema operativo". In particolare negli ultimi anni vengono portati a confronto sistemi diversi come Linux e Windows e per entrambi si utilizza la stessa etichetta di "sistema operativo" (SO). Ovviamente le problematiche sono le stesse, o per lo meno molto simili, almeno al livello più vicino all'hardware; ma non appena si risale lungo la pila di astrazioni necessarie a costruire un sistema completo utilizzabile dall'utente, sorgono immediatamente differenze concettuali di grande rilievo.

Esistono diverse correnti di pensiero su cosa sia un sistema operativo e quali i compiti affidatigli. Innanzitutto va compreso il fatto che, ancor più dei sistemi naturali, quelli informatici sono costituiti da una pila di astrazioni concettuali, ciascuna delle quali si poggia sulla precedente. Al livello più basso, vi sono i

semiconduttori (e forse a un livello precedente vi è la meccanica quantistica); ai livelli successivi vi sono i circuiti logici, i microprocessori, i sistemi microelettronici e quindi quello che è più familiarmente noto come “hardware” di un computer. Tutti questi componenti rimangono pur sempre “hardware”, roba dura, componenti materiali, che continuano ad esistere anche se si toglie la corrente e si passa per un campo magnetico di un milione di Tesla, e che sono sostituibili solo buttando un pezzo solido e mettendone un altro analogo al suo posto. In termini ancora più semplici, “le parti che puoi colpire con un martello sono l’hardware; quelle contro cui puoi solo inveire sono il software” (anonimo).

In maniera più assertiva, il software può essere definito come una sequenza di istruzioni che l’hardware è in grado di eseguire. Nonostante in alcuni casi non sia evidente, anche il software è costruito su pile di livelli. Questo concetto, sicuramente familiare agli utenti Linux, dovrebbe essere chiaro anche agli utenti Windows con un esempio: se vi dotate di una nuova periferica, per esempio un masterizzatore, dovrete innanzitutto installare il “driver” di quella periferica e quindi potrete installare il programma che la utilizza. A volte questa stratificazione è nascosta dalla procedura di installazione, ma in generale si tratta di qualcosa di assai concreto: in caso contrario, non potrebbero esistere programmi generici (per esempio un programma di masterizzazione) che funzionino con una gran varietà di dispositivi diversi (tutti i diversi modelli di masterizzatori in circolazione).

I driver per periferiche sono spesso considerati parte del sistema operativo, ma non al livello più basso di astrazione; a loro volta, infatti, debbono interagire con la memoria RAM e con il processore. Il software che gestisce queste due componenti è il

“nucleo” (“kernel”) del sistema operativo. Comunque anche il termine “kernel” è utilizzato in maniera ambigua: il kernel Linux, così com’è distribuito sul sito dedicato (www.kernel.org), comprende anche una pleora di device driver, ma gli sviluppatori del kernel li considerano in maniera ben distinta dal “kernel vero e proprio” (“the kernel proper”). Analogamente, altri elementi vengono considerati a un livello più elevato del kernel vero e proprio, come i protocolli di comunicazione, i file system, le funzioni di sicurezza – componenti senza le quali nessun computer potrebbe funzionare. In pratica, nell’uso più largamente diffuso, il “kernel” di un sistema operativo è quella parte di software che interagisce direttamente con l’hardware *più* una parte di software che fornisce una serie di astrazioni utili a semplificare la scrittura del resto del software. Sulla base di questa definizione, è possibile che il kernel di un sistema presenti una serie di astrazioni differenti rispetto a un altro sistema; come nel caso di Windows, le cui astrazioni grafiche non esistono invece in Linux.

Quanto al resto del software, la distinzione è ancora più vaga: “sistema operativo” viene generalmente considerato tutto ciò che viene fornito sugli stessi supporti su cui viene fornito il kernel. Tutto ciò per cui è necessario ottenere licenze aggiuntive, o che comunque non viene fornito insieme al sistema operativo, viene classificato tra le “applicazioni di terze parti”, o “applicazioni” in breve. Windows, ad esempio, è un sistema operativo molto scarno, in quanto contiene solo una serie alquanto basilare di applicazioni (che comunque con il tempo si è arricchita, apparentemente per consentire a Microsoft di ampliare il proprio monopolio). Invece un sistema operativo basato su Linux normalmente contiene una vasta gamma di applicazioni, tanto che non ha senso installarle tutte.

Ma la nostra storia ci porta molto più indietro nel tempo, quando tutte le distinzioni di cui sopra non esistevano ancora. I primi computer erano poco più di grandi calcolatrici: i programmi venivano scritti da utenti estremamente specializzati, i tecnici si occupavano di caricare e scaricare manualmente, mentre il calcolatore non doveva far altro che seguire le istruzioni e stamparne i risultati. Ciò andava bene finché si ricorreva ai calcolatori per la precisione offerta, più che per la velocità: quando la tecnologia raggiunse il livello in cui una macchina doveva rispondere alle esigenze di decine o centinaia di utenti, il caricamento manuale dei programmi diventò operazione decisamente penosa. Ecco allora la nascita dei primi sistemi a condivisione di tempo (“time sharing systems”), di cui Unix costituisce uno dei primi e migliori esempi. In pratica, la responsabilità principale del sistema operativo (o, meglio, del kernel) riguarda la gestione dell’accesso alle risorse, in primo luogo al processore e alla memoria, ma anche alle periferiche, da parte di alcune decine o centinaia di programmi. Il compito è più corposo di quanto appaia dalla semplice descrizione, e sotto certi aspetti le tecniche per svolgerlo nella maniera migliore sono tuttora oggetto di studio.

Lo Unix tra gli Un*x

Sebbene nella scala dei tempi informatici ciò risalga alla preistoria, è istruttivo iniziare la storia di Unix proprio dal nome. Dietro il quale, del resto, non vi è molto di più di quanto appaia evidente: una contrazione dell’aggettivo “unico”, “unique” in inglese. Questa scelta era, nell’anno di nascita 1969, null’altro che un “insider joke”, un gioco di parole per specialisti: il sistema operativo multitasking sviluppato congiuntamente da

AT&T, General Electrics e il MIT tra il 1965 e il 1968 si chiamava MULTICS, acronimo per “MULTiplexed Information and Computing Service”, ma anche come gioco di parole per richiamare la “molteplicità” del sistema. Il passaggio dalla molteplicità di MULTICS all’unicità di Unix voleva simboleggiare il passaggio dalla complessità alla semplicità, dall’attenzione verso la quantità di caratteristiche a quella per l’integrità concettuale, dalle esigenze commerciali a quelle di efficienza nell’uso delle risorse.

MULTICS soffriva di quella che in gergo viene chiamata “sindrome da secondo sistema”, popolarizzata da Frederick P. Brooks: costruito sulla base del “Compatible Time Sharing System”, implementava tutte le caratteristiche che i suoi creatori avrebbero voluto fossero presenti nel predecessore. Eric S. Raymond ha teorizzato che Unix sia il risultato di un “effetto da terzo sistema”: quando si crea qualcosa di innovativo, il primo sistema è spesso poco più di una dimostrazione della fattibilità dell’idea, il secondo crolla sotto il peso di tutte le caratteristiche che si vogliono aggiungere e il terzo finalmente viene sviluppato correttamente. Considerando che MULTICS è comunque sopravvissuto fino al 2000, c’è da credere che Unix, comunque vadano le cose, ci accompagnerà quanto meno per buona parte del ventunesimo secolo.

Il primissimo Unix del 1969 in realtà si chiamava UNICS, in contrapposizione ancora più spinta con il MULTICS. Ancora oggi, il termine UNIX, scritto in maiuscole, è un marchio registrato di The Open Group. Per motivi di maggiore scorrevolezza, il termine “Unix” verrà indicato in seguito con la sola iniziale maiuscola, come da usanza diffusa nella comunità open source, per riferirci a tutti i sistemi derivati da UNIX o scritti

con l'intento specifico di somigliargli. La forma UNIX verrà utilizzata solo nei punti in cui si fa riferimento esplicito al marchio registrato.

La genesi di Unix

Il luogo di nascita di Unix sono i mitici Laboratori Bell ("Bell Labs") della AT&T, una struttura di ricerca industriale creata nel 1907 e nella quale sono state compiute alcune delle più grandi scoperte industriali del secolo, dal tubo a raggi catodici al transistor, dal laser al radar, e nel quale sono stati ricevuti ben cinque premi Nobel. Nel 1969, la AT&T decise di abbandonare il progetto MULTICS, principalmente a causa della divergenza di obiettivi con i partner e della lentezza con cui procedeva il progetto. Nel tempo MULTICS è divenuto un prodotto commerciale di un certo successo ed è rimasto in uso fino al 2000, ma all'epoca i progressi erano stati così scarsi da rendere la situazione scoraggiante per i ricercatori dei Bell Labs: il sistema, progettato per poter gestire contemporaneamente fino a mille utenti, funzionava adeguatamente solo con tre.

Fu un gruppo del tutto informale di ricercatori dei Bell Labs, disperato per la mancanza di un ambiente di calcolo adeguato su cui operare, che prese l'iniziativa di proporre l'avvio di un progetto di sviluppo per un nuovo sistema operativo. Il coordinatore ufficioso del progetto di Unix era Ken Thompson, il quale voleva creare un ambiente che comprendesse le migliori caratteristiche di MULTICS: il concetto di *elaborazione*, un *file system* ad albero, un'interfaccia utente basata su un interprete di comandi, un *file di testo* per la gestione dei dati e accesso generalizzato ai dispositivi. In effetti, nonostante i problemi in cui si dibatteva il progetto MULTICS, i ricercatori dei Bell Labs

ne apprezzavano parecchie delle caratteristiche potenziali: alla base di Unix c'era soprattutto il desiderio di sviluppare un ambiente adeguato in cui lavorare dopo l'abbandono di MULTICS. Nelle parole di Thompson, "volevamo salvare non solo un buon ambiente di programmazione, ma un sistema intorno al quale potesse formarsi una comunità".

Altro nome destinato a passare alla storia fu quello di Dennis Ritchie, creatore del linguaggio di programmazione C, popolarizzato in un manuale, da lui scritto insieme a Brian Kernighan, che è ancora oggi la migliore introduzione alla programmazione in C. Inizialmente Unix era scritto in linguaggio Assembler, il quale è poco più di una codifica mnemonica dei codici numerici delle istruzioni macchina che un computer è in grado di capire; il C introduce un livello di astrazione minimo rispetto alle modalità di lavoro del computer, ma sufficiente a introdurre i costrutti di base della programmazione strutturata. L'introduzione del C è così importante da costituire una delle ragioni fondamentali di oltre trent'anni di successo dei sistemi unixoidi.

Le prime elucubrazioni su Unix avvenivano nel 1969, anno in cui Thompson riusciva anche ad avere tempo libero a sufficienza per creare uno dei primi giochi elettronici, "Space Travel", che simulava un viaggio spaziale attraverso il sistema solare. Per poter sviluppare e utilizzare il gioco a costi ragionevoli, Thompson aveva recuperato un computer considerato obsoleto e poco utilizzato, un PDP 7, che quindi divenne in maniera abbastanza naturale il primo computer su cui venne provata la primissima versione di Unix. Nel 1970 il progetto aveva già raggiunto risultati tali da convincere la dirigenza dei laboratori all'acquisto di un computer dedicato allo sviluppo del nuovo sistema, un PDP 11, dotato di ben 24 KB di memoria

(una frazione di quella di cui sono dotate le attuali SIMM GSM) e inizialmente privo di dischi. Il 1971 vedeva l'arrivo dei primi utenti Unix, il dipartimento brevetti dei Bell Labs. Nello stesso periodo iniziava lo sforzo per riscrivere il sistema in un linguaggio di più alto livello. Sembra che Thompson abbia tentato il Fortran per una giornata, per poi passare immediatamente a creare il proprio linguaggio, chiamato B. Rapidamente Ritchie vi aggiunse i tipi astratti, creando il cosiddetto "New B" o NB, arrivando così nel 1972 alla riscrittura completa del linguaggio, che venne battezzato "C". Nell'estate del 1973 nasceva Unix come lo conosciamo oggi, completamente riscritto in C.

Da allora si sono susseguiti innumerevoli migliorie e cambiamenti: l'albero delle versioni di Unix (alcune delle quali hanno ben poco in comune tra loro) è estremamente complesso e intricato, soprattutto se esaminato a livello di codice. Altri aspetti sono però universalmente presenti in tutte le versioni, e ne costituiscono l'aspetto più interessante, come la logica progettuale, e sono stati sintetizzati in una serie di aforismi. I più importanti sono forse quelli di Doug McIroy, direttore del dipartimento dove lavorava Thompson nonché inventore delle "pipe", centrali nella logica Unix: "Scrivere programmi che fanno una cosa sola e la fanno bene. Scrivere programmi che possano lavorare insieme. Scrivere programmi che gestiscano flussi di testo, che costituisce un'interfaccia universale". Questi sono i principi esprimono la logica alla base stessa di Unix, ben più dei dettagli di codifica.

L'esodo

La nascita di Unix è dunque legata al mondo della ricerca, all'interno di una delle realtà industriali più importanti del boom

economico statunitense. A quei tempi ben difficilmente la ricerca industriale diventava liberamente disponibile al di fuori dell'ambito da cui nasceva. Ma allora AT&T poteva fornire solo servizi di telefonia, a seguito dell'accordo con cui aveva chiuso una causa Antitrust con il governo statunitense nel 1956. Così Thompson e i suoi collaboratori erano liberi di presentare i risultati raggiunti nell'editoria e ai convegni di settore, e l'invio di copie dei codici sorgenti di Unix poteva essere effettuato, almeno nei primi anni, senza troppe formalità e senza le licenze sul codice sorgente che AT&T avrebbe iniziato a far pagare dopo qualche anno.

A seguito della prima presentazione pubblica di Unix nel 1973, nel giro di sei mesi il numero di installazioni passò da 16 a 48. Nel 1974 uscì sulla rivista specializzata *Communications of the ACM* il noto articolo "The UNIX Time-Sharing System" a firma Thompson e Ritchie, che diffuse la fama del sistema operativo in tutto il mondo e fece perdere rapidamente il conto delle installazioni: gli invii di nastri di Unix dai Bell Labs raggiunsero rapidamente le centinaia di unità l'anno. Con un costo non inferiore ai 50.000 dollari per un qualunque mini-computer, il mercato dei "personal computer" e del software era ancora molto al di là da venire. Le limitazioni sulla distribuzione del software, soprattutto nell'ambito della ricerca, erano del tutto sconosciute.

Le università che ricevevano i nastri di Unix non si limitavano a utilizzarli così com'erano, ma vi apportavano aggiunte e migliorie: adattamento a diversi computer, supporto per nuove periferiche, correzione di bachi nel software, introduzione di nuove idee e migliorie, scrittura da zero di nuovi programmi utilizzabili in ambiente Unix. Nelle mani di Thompson,

prende forma una raccolta di codice sorgente in cui i contributi provenienti dall'esterno divenivano via via più importanti. In molti casi tali contributi, pur validi dal punto di vista concettuale, richiedevano una riscrittura interna ai Bell Labs per migliorarne la realizzazione pratica. È il caso di un esempio molto importante, l'algoritmo di allocazione della memoria. A Berkeley era stata realizzata un'implementazione di un algoritmo, in seguito divenuto un classico dell'informatica, pubblicato su *The Art of Computer Programming* di Donald E. Knuth, come dire la bibbia dell'informatica: il codice inviato a Thompson per l'inclusione in Unix venne in realtà riscritto da zero da Dennis Ritchie.

L'Università della California a Berkeley (UCB) era stata una delle prime università ad installare Unix V4. Il successo del nuovo sistema fu tale da rendere necessario, nel 1974, l'acquisto di un secondo PDP, dedicato al progetto di database INGRES, che a fine anno avrebbe iniziato a distribuire autonomamente su nastro il proprio software, insieme al sistema operativo. Nel 1975 venne acquistato un terzo computer, che fu consegnato in coincidenza con l'arrivo a Berkeley di Ken Thompson, in anno sabbatico. Periodo in cui, oltre a insegnare Unix agli studenti, Thompson avrebbe anche rilasciato e installato sul nuovo computer la versione 6 (V6) di Unix.

Nell'autunno del 1975 si immatricolava a Berkeley anche un allora sconosciuto Bill Joy, che sarebbe divenuto una delle figure più influenti della cultura Unix. Inizialmente Joy si dedicò a utilizzare e migliorare il compilatore Pascal creato da Thompson e a realizzare un editor di testi migliore di quelli disponibili. Prese a scrivere, insieme al collega Chuck Haley, l'editor "ex", che lavorava sulle singole linee di testo. Le caratteristiche

dei terminali disponibili consentivano l'uso di un editor che lavorasse a schermo intero, ma Unix non ne era ancora dotato. Così nel 1977 Joy introdusse "vi", ancora oggi uno degli editor preferiti degli appassionati Unix. Per renderlo utilizzabile sui diversi sistemi dell'Università, Joy dovette creare "termcap", una libreria di astrazione delle caratteristiche dei terminali che è ancora utilizzato in tutti i sistemi Unix.

Nel corso dello stesso anno Joy creava la "Berkeley Software Distribution", una collezione di programmi utilizzabili in ambiente Unix. La seconda versione, "2BSD", avrebbe visto la luce nel 1978 e l'anno seguente sarebbe iniziata la distribuzione della versione 3, completamente riadattata per girare anche su sistemi VAX. Modifica particolarmente importante e difficile, perché il VAX aveva uno spazio di indirizzamento di 32 bit, rispetto ai 16 bit della serie PDP: questo lo rendeva estremamente più potente dei suoi predecessori, ma anche più complesso da programmare. La DEC, produttrice sia dei PDP che del VAX, ha vanamente tentato per decenni di scoraggiare l'uso di Unix sulle proprie macchine allo scopo di spingere il proprio sistema operativo proprietario.

Nel frattempo andavano preparandosi eventi che si sarebbero rivelati fondamentali per il futuro di Unix e per l'affermazione di Berkeley come maggior centro per lo sviluppo e la distribuzione di software Unix. L'Agenzia per i progetti di ricerca avanzata per la Difesa ("Defense Advanced Research Projects Agency", ovvero DARPA) aveva realizzato ARPAnet, una rete nazionale di collegamenti tra i computer dei principali centri di ricerca; l'evoluzione naturale di questa rete diede poi vita all'attuale Internet. Verso la fine degli anni Settanta, iniziava a manifestarsi la necessità di sostituire i computer interconnessi

da ARPAnet. Sebbene i protocolli di comunicazione utilizzati fossero stati progettati in modo da usare sistemi operativi differenti, non pareva praticabile richiederne l'implementazione ai vari rivenditori hardware, così come non sembrava opportuno scegliere un unico fornitore. Venne così presa la decisione di unificare l'infrastruttura a livello di sistema operativo, oltre che a quello di protocollo di comunicazione, e la scelta cadde su Unix, del quale era già stata ampiamente dimostrata la portabilità e che prometteva quindi di poter essere utilizzato su piattaforme hardware assai diverse.

Nel 1980 Bob Fabry, docente presso la stessa Università, otteneva una sovvenzione dal DARPA per l'implementazione dei protocolli TCP/IP in Unix. Il rilascio di 3BSD, nel dicembre '79, aveva avuto un peso non piccolo nella decisione di DARPA perché dimostrava praticamente come Berkeley fosse del tutto in grado di produrre un sistema completo funzionante. Il gruppo formato da Fabry prese il nome di "Computer Systems Research Group", ovvero CSRG. Il coordinatore dello sviluppo software era Bill Joy, che per accettare la posizione aveva rinunciato al dottorato di ricerca. I risultati del CSRG, concretizzati in 4.1BSD a metà del 1981, furono tali da convincere DARPA a rinnovare la sovvenzione iniziale di 18 mesi per altri due anni. Le versioni successive, la 4.2 del 1983 e la 4.3 del 1986, ebbero un successo di distribuzione impressionante. Nel frattempo Bill Joy aveva lasciato il progetto per dedicarsi alla commercializzazione di Unix, divenendo uno dei membri fondatori di Sun Microsystems.

Negli anni successivi, BSD venne lentamente liberato da tutto il codice proveniente da AT&T. Inizialmente, secondo la ricostruzione di Marshall Kirk McKusick, l'obiettivo era pura-

mente pragmatico: molte organizzazioni volevano utilizzare il codice di networking del CSRG senza essere costrette a ottenere prima una licenza per il codice sorgente dalla AT&T. La situazione era aggravata dal fatto che nel 1983 la AT&T, dopo aver perso una nuova causa per abuso di posizione dominante, fu costretta a rinunciare al servizio telefonico locale, mantenendo quello a lunga distanza. Le sette compagnie (familiarmente dette “Baby Bells”) nate dalla suddivisione di AT&T per gestire il traffico telefonico locale nelle varie zone degli Stati Uniti, si erano ridotte a cinque in seguito a successive fusioni (Bell Atlantic, SBC, Bell South, Ameritech, U.S. West). Il risultato poteva essere positivo per il mercato telefonico, ma si dimostrò quasi fatale per Unix (lo vedremo meglio nella prossima sezione).

Dapprima il CSRG si limitò a estrarre tutta la parte di codice di networking di cui possedeva il copyright – e che quindi poteva ridistribuire a proprio piacimento – e a renderlo disponibile sotto un nome non molto fantasioso (“Networking Release 1” o Net/1 in breve) e con una licenza assai aperta. Il codice poteva essere ridistribuito, modificato o non modificato, in formato sorgente o solo eseguibile, senza dover pagare costi aggiuntivi all’Università della California. L’unica restrizione concerneva il divieto di rimuovere le note di copyright dal codice sorgente e l’obbligo di indicare nella documentazione dei prodotti derivati che era stato utilizzato software realizzato dalla stessa Università.

Successivamente il nuovo coordinatore dello sviluppo software del CSRG, Keith Bostic, propose la riscrittura completa di tutte le parti provenienti dalla AT&T. Per rendere abordabile l’impresa, Bostic sperimentò una nuova tecnica di sviluppo:

pubblicizzò le sue intenzioni via rete, incoraggiando chi volesse partecipare a riscrivere i programmi necessari, con il solo incoraggiamento di avere il proprio nome nell'elenco degli sviluppatori. Il riscontro fu entusiasta: nel giro di pochi mesi il CSRG poté concentrarsi solo sul kernel e nel giugno del 1991 veniva distribuita la "Networking Release 2" (Net/2). Non si trattava di un sistema compilabile perché mancavano sei file di header non semplicemente riscrivibili. Nell'arco di pochi mesi, naturalmente, la comunità Unix chiuse il buco: Bill Jolitz, dopo aver riscritto i file mancanti, rese disponibile su Internet la prima versione di 386/BSD. I suoi sforzi furono raccolti dapprima dal progetto NetBSD, dedicato all'adattamento di BSD al maggior numero possibile di piattaforme hardware diverse, e in seguito da FreeBSD, focalizzato sulla piattaforma PC x86. In seguito a uno scisma interno alla comunità FreeBSD, è nato infine il progetto OpenBSD, che per darsi una caratterizzazione ha deciso di dedicare particolare attenzione alla sicurezza e alla stabilità.

L'apocalisse

A partire dalla fine degli anni Settanta, ma soprattutto nel corso degli anni Ottanta, Unix raggiunse il mondo industriale con una diffusione senza precedenti per un sistema operativo. Le ragioni di questo successo sono duplici: da un lato, l'elevata portabilità del sistema, che già nel 1979 ne aveva consentito il rapido adattamento da PDP a VAX; dall'altro, l'arrivo sul mercato delle "workstation", una nuova categoria di computer, successore verso il basso dei precedenti "minicomputer" come il PDP o il VAX. Così come l'arrivo del VAX aveva portato nei dipartimenti universitari risorse di calcolo prima inavvicinabili, le workstation le estesero al mondo commerciale, spingendolo-

ne ancor più la diffusione in università e centri di ricerca. Almeno tre i vantaggi fondamentali offerti dall'uso di Unix per i produttori di hardware: l'ampia base di codice disponibile riduceva i costi di sviluppo del sistema operativo; i programmatori neo-laureati sempre più spesso avevano imparato a operare in ambiente Unix all'università; e, soprattutto, la portabilità di Unix si traduceva non solo nella possibilità di riadattare il sistema operativo a un nuovo ambiente, ma anche a un veloce riadattamento dei programmi da uno Unix a un altro. Così, gli anni Ottanta, videro la nascita di decine di nuovi Unix, mirati a specifici sistemi hardware, come Sun/OS, HP/UX, Ultrix, Unicos, Xenix, MIPS OS, AIX, Mach, Minix, Plan 9, NeXT-STEP – per citare solo alcuni dei più noti.

Come descritto sopra, dal 1984 la AT&T, essendo stata costretta a rinunciare al mercato della telefonia locale, non soggiaceva più ai termini dell'accordo bonario del 1956. Venne quindi creata una società consociata per il mercato informatico, la "AT&T Computer Systems", il cui software di bandiera era "Unix System V". L'ingresso del colosso della telefonia in un mercato nuovo, quello dei sistemi basati su Unix, inizialmente non ebbe grande impatto. In effetti, data la natura aperta di Unix, l'unico vantaggio di AT&T nei confronti della concorrenza era la proprietà intellettuale del codice sorgente, e quindi l'assenza di alcuna necessità di acquistare licenze. Per lo stesso motivo, Unix rimaneva per AT&T una fonte di introiti: chiunque utilizzasse i codici sorgenti di Unix doveva pagare ad AT&T il costo della licenza corrispondente, indipendentemente dal fatto che, come base di partenza, fossero stati utilizzati i codici dell'Università della California piuttosto che quelli dei Bell Labs.

La contrapposizione tecnica tra i due sistemi assumeva spesso i termini di “System V style” rispetto a “BSD style”, ma gli aspetti tecnici erano solo la punta dell’iceberg del confronto-scontro in atto. Il mercato Unix era del tutto estraneo ad AT&T e si giocava su dettagli quali le caratteristiche della piattaforma hardware, le migliorie effettuate sul codice di Unix, le applicazioni che venivano fornite, la qualità del supporto e dell’assistenza. Ambiti questi in cui AT&T era stata sostanzialmente sempre assente: Unix veniva considerato soltanto strumento di supporto alla gestione di sistemi telefonici. Non a caso la licenza sul codice sorgente includeva esplicitamente clausole di esclusione di qualsiasi tipo di responsabilità e di assistenza da parte di AT&T. Con l’effetto collaterale di costringere i rivenditori di sistemi Unix a farsi un’esperienza molto maggiore di quella esistente nella stessa AT&T. Di fatto, in un mercato così difficile e competitivo, AT&T aveva un unico vantaggio: la proprietà intellettuale sul codice sorgente di Unix, che però fruttava assai meno di quanto l’azienda desiderasse.

Non deve quindi meravigliare il fatto che la nuova società decidesse di allearsi con Sun Microsystems, la società fondata da Bill Joy, Scott McNealy, Vinod Khosla ed Andy Bechtolsheim, rapidamente impostasi come una delle più solide realtà commerciali costituitesi intorno a Unix. In pratica, AT&T ne guadagnava un partner esperto del mercato Unix e Sun otteneva un vantaggio competitivo sulla concorrenza dal punto di vista della gestione delle licenze sul codice sorgente. Ovviamente le altre società del mercato Unix non restarono con le mani in mano. Nel 1988 alcune di queste, tra cui IBM, DEC e HP, fondarono la “Open Systems Foundation” (OSF), consorzio dedicato alla creazione e alla promozione di uno standard Unix aper-

to. La risposta di AT&T fu tanto scontata quanto distruttiva: la creazione di un'analoga cordata, "Unix International" (UI), avente finalità e obiettivi identici. Il complesso e intricato conflitto in atto tra le due fazioni, di cui la nascita di OSF e UI rappresentava solo la punta dell'iceberg, venne battezzato dalla stampa come "le guerre Unix" ("Unix wars"). La cultura hacker, radicata nelle università, si trovava ovviamente dalla parte di BSD, mentre AT&T, ben prima di Microsoft, fu l'archetipo dell'azienda "cattiva" da odiare, al punto da essere identificata con la "Morte Nera" di *Star Wars*, anche grazie alla somiglianza del simbolo AT&T con il satellite infernale del film di Lucas. Eric S. Raymond, esponente della cultura hacker che sarebbe divenuto uno dei fondatori del movimento Open Source, scrisse "Unix Wars", parodia della trama di *Star Wars*.

Può sembrare paradossale che nello stesso anno in cui nascevano UI e OSF venisse pubblicato uno standard che in teoria avrebbe dovuto renderne del tutto inutile l'esistenza. Il successo commerciale di Unix si era dimostrato un boomerang per le aziende governative: dopo aver fatto sviluppare Unix come sistema indipendente dall'hardware che gestisse i protocolli di comunicazione di ARPAnet, si trovavano a dover fronteggiare una pletera di implementazioni commerciali del sistema, senza un riferimento che stabilisse quali fossero i requisiti minimi perché il sistema potesse essere considerato "come Unix". Per ovviare a questo problema, la IEEE ("Institute of Electrical and Electronics Engineers", associazione internazionale tecnica professionale non a fini di lucro) condusse uno studio approfondito sulle caratteristiche di Unix, dettagliandole in un corposo lavoro che prese il nome di POSIX ("Portable Open System Interface"; la *x* è un'aggiunta in tipico stile Unix). La

pubblicazione dello standard avvenne proprio nel 1988, e da allora è rimasto punto di riferimento per aziende governative e società commerciali che volessero richiedere un sistema “di tipo Unix” in maniera obiettiva e “super partes” – qualcosa di particolarmente importante visto il clima delle guerre Unix in corso. POSIX ha anche giocato un ruolo fondamentale nello sviluppo di Linux, come vedremo più avanti.

La battaglia decisiva delle guerre Unix fu scatenata dal rilascio di Net/2 da parte del CSRG: a quel punto, esistevano di fatto due versioni complete e indipendenti di Unix, utilizzabili come base per un nuovo sistema. È vero che il prodotto del CSRG non era completo e compilabile, ma altrettanto vero è che mancavano pochi file, quelli presenti avevano un’interfaccia chiara e dipendevano dall’architettura hardware di appoggio, per cui andavano comunque riscritti da ciascun produttore. Tra l’altro, AT&T era probabilmente molto meno consapevole delle condizioni del proprio codice sorgente di quanto non lo fosse l’Università della California. Prima di procedere al rilascio della Net/2, il codice era stato dettagliatamente verificato, per cui vi era una ragionevole certezza che nessuno dei file contenuti fosse stato scritto nei Bell Labs. Viceversa, poiché lo scopo di AT&T non era mai stato quello di commercializzare Unix, l’introduzione dei codici provenienti dall’esterno nello sviluppo di Unix System V era avvenuto in maniera assai meno controllata. Per lo stesso motivo, la pratica di inserire note di copyright nei file della AT&T aveva preso corpo molto tardi rispetto alle prime versioni: gran parte del codice originale si trovava così ad essere stato distribuito come software di pubblico dominio. Immediatamente dopo il rilascio della Net/2, il CSRG formò la “Berkeley Software Design, Incorporated” (BSDI), una

società dedicata alla commercializzazione del nuovo sistema. Si trattava della naturale evoluzione della vendita dei codici della Net/1, in precedenza condotta direttamente dal CSRG, pur se con alcune importanti differenze. All'inizio del 1992, la BSDI iniziò una campagna pubblicitaria per la vendita del sistema, sia in formato eseguibile che sorgente, che veniva chiamato UNIX ed era esplicitamente paragonato a UNIX System V, rispetto al quale era "offerto uno sconto" del 99 per cento. Era anche stato registrato un numero verde che, con la tipica trasformazione in lettere era diventato 1-800-ITS-UNIX, tanto per togliere ogni dubbio sul prodotto di cui si stava parlando. La campagna sotto questa forma durò pochissimo perché gli "Unix System Laboratories" (USL), la divisione di AT&T che si occupava della commercializzazione di Unix, richiesero immediatamente alla BSDI la rinuncia al nome Unix e al numero di telefono. La BSDI, riconoscendo una causa persa, corresse la campagna pubblicitaria e rinunciò al numero, ma gli USL proseguirono la battaglia con una citazione in tribunale in cui si sosteneva che il codice BSDI conteneva materiale proprietario e segreti commerciali degli USL. La parte sui "segreti commerciali" può apparire particolarmente sfacciata, riferita a un prodotto con la diffusione e la notorietà di Unix, distribuito con i codici sorgenti fin dall'origine. D'altra parte, l'unico modo di richiedere al giudice un'ingiunzione per l'immediato ritiro dal commercio del prodotto BSDI era sostenere che gli USL avrebbero subito danni irreparabili per il proseguimento delle vendite: per sostenere questa tesi pareva non esistesse altro modo che la via dei "segreti commerciali" disseminati nel codice.

Alla prima udienza, BSDI fece presente che il suo prodotto era

basato sul software dell'Università della California, cui erano stati aggiunti solo sei file. Gli USL si videro quindi costretti a riformulare la richiesta di ingiunzione citando sia BSDI che l'Università ed estendendola anche alla distribuzione Net/2. All'inizio del 1993, il giudice federale negò la richiesta di ingiunzione e suggerì di discutere il processo in un tribunale statale, anziché seguendo l'iter giudiziario a livello federale. L'Università della California si affrettò a depositare per prima la citazione, in modo da poter proseguire la causa in un tribunale locale, più vicino e probabilmente più favorevole. Nel testo dell'istanza, gli USL venivano accusati di non aver ottemperato all'obbligo di riconoscere il contributo dell'Università per l'utilizzo del codice di BSD in System V.

Tuttavia il dibattimento non approdò in aula, perché nel giugno dello stesso anno gli USL vennero acquistati da Novell, il cui amministratore delegato, Ray Noorda, espresse subito la volontà di trovare un accordo amichevole. Dopo quanto era successo, le discussioni non potevano rivelarsi troppo facili, ma all'inizio del 1994 si trovò finalmente un punto d'incontro: la UCB decise di rimuovere tre file (su un totale di 18.000), apportare lievi modifiche ad altri file e aggiungere le note del copyright USL ad altri ancora, i quali rimanevano però liberamente distribuibili.

Verso la fine del '93 Novell trasferì la proprietà del marchio UNIX a X/Open, un consorzio di industrie costituito da Bull, ICL, Siemens, Olivetti e Nixdorf (inizialmente noto anche come BISON, cioè "bisonte"). Nel 1996 X/Open si fuse con la OSF, costituendo "The Open Group", che a oggi detiene ancora il controllo del marchio UNIX. Quanto al codice sorgente di Unix, dovrebbe essere passato da Novell a SCO nel 1995, insieme alla

propria divisione UnixWare. Ma come vedremo nei capitoli successivi, è d'obbligo il condizionale, dal momento che Novell ha citato in giudizio SCO proprio su questo punto.

Il GNUovo Testamento

Unix compie quest'anno 35 anni: una veneranda età per un sistema operativo, sia pure in incarnazioni completamente diverse da quelle originali. I suoi contemporanei sono estinti, per cause simili a quelle per cui si estinguono le specie viventi: l'impossibilità di adattarsi ad un nuovo ambiente. Un esempio di specie contemporanea di Unix che si è estinta, piuttosto brutalmente, per mancanza di adattamento è l'“Incompatible Time Sharing System” (ITS). Questo sistema era stato sviluppato nel Laboratorio di intelligenza artificiale (“Artificial Intelligence” o AI, come nel recente film di Spielberg) del glorioso Massachusetts Institute of Technology (MIT), a Cambridge, nella città di Boston. Il Laboratorio AI era una fucina di menti brillanti nel campo della tecnologia informatica, che si consideravano parte della cultura “hacker”, e l'ITS era stato messo a punto per utilizzare al meglio la serie di computer PDP. Come e più di Unix, l'ITS rappresentava un sistema brillante e innovativo; e, al pari di Unix, venne così chiamato in esplicita antitesi a un sistema che voleva sostituire e migliorare, il “Compatible Time Sharing System” o CTSS. L'ITS è considerato dalla cultura hacker una propria creatura, anzi una delle realizzazioni più riuscite.

Gli hacker del laboratorio AI avevano costruito intorno all'ITS una cultura analoga a quella formatasi oggi intorno a sistemi quali Linux o i vari BSD, sebbene ovviamente limitata a una

singola sede e situazione. Il pensionamento dell'ultimo PDP-10, nel 1982, costituì un primo duro colpo per quel gruppo: riscrivere un sistema analogo all'ITS per il nuovo calcolatore, un mainframe KL-10 con sistema operativo TOPS-20, era pressoché impossibile, sia perché il nuovo computer non serviva solo il laboratorio AI ma un intero dipartimento, sia perché buona parte della comunità hacker ormai lavorava fuori dal laboratorio, pur se non a tempo pieno. Dal lavoro interno sul Lisp, linguaggio particolarmente adatto alle ricerche sull'intelligenza artificiale, erano nate due aziende, la Symbolics e la LMI, entrambe con l'obiettivo di commercializzare un calcolatore basato su Lisp e con gli stessi hacker del laboratorio in qualità di consulenti. Le due aziende iniziarono così a combattere una battaglia commerciale non dissimile dalle successive guerre Unix, sebbene su scala ridotta e in un settore che si sarebbe poi rivelato una bolla di sapone. Il risultato fu il completo disfacimento del gruppo di hacker del laboratorio AI.

Il 27 settembre del 1983, sul newsgroup Usenet *net.unix-wizards*, apparve un annuncio che sarebbe passato alla storia di Unix:

[...]

A partire dal prossimo giorno del Ringraziamento inizierò a scrivere un sistema compatibile con Unix chiamato GNU (Gnu's Not Unix), distribuito gratuitamente a chiunque vorrà usarlo. Urgono contributi in tempo, denaro, programmi e attrezzature.

[...]

Il messaggio proveniva da Richard M. Stallman (noto ai colle-

ghi con le semplici iniziali RMS), un hacker del gruppo del laboratorio d'intelligenza artificiale. Dotato di grande talento per la programmazione, un carattere originale e un senso etico estremamente rigido, Stallman aveva provato a opporsi per quanto possibile all'invasione del laboratorio da parte del mondo aziendale. Non riuscendovi, si era trovato a scegliere tra abbandonare la passione per l'informatica, lasciar cadere ogni obiezione verso il software proprietario oppure dedicarsi alla creazione di un sistema alternativo, non proprietario. Optando per quest'ultima alternativa, Stallman si è poi rivelato il vero e proprio profeta della comunità del software libero/open source.¹

Il progetto di Stallman è partito in un certo senso dall'esterno verso l'interno, cioè dai programmi di utilità verso il kernel del sistema. Un'esigenza che ha profondamente condizionato la lentezza dello sviluppo del sistema GNU è stata quella di conciliare l'ampiezza del progetto con la necessità di ottenere una certa visibilità e di iniziare ad avere un qualche margine economico. Di conseguenza, fin dall'inizio il progetto GNU ha diffuso un prodotto che andava adattato a sistemi Unix, l'editor Emacs, creato da Stallman durante il periodo al laboratorio AI. Definire Emacs un editor di testi è in realtà quasi limitante: utilizzato pienamente, il programma si dimostra un ambiente di lavoro completo, essendo dotato di caratteristiche per la gestione della posta elettronica, dei newsgroup Usenet, della navigazione web, della shell di sistema e di decine di altre caratteristiche. I detrattori di Emacs hanno classicamente messo l'ac-

¹ Per maggiori dettagli sulle attività e sul pensiero di Richard Stallman, si veda la sua biografia "ufficiosa" curata da Sam Williams (Codice Libero: La crociata di Richard Stallman a sostegno del software libero, Apogeo, 2003), nonché i due volumi italiani di saggi scelti dello stesso Stallman (Software Libero, Pensiero Libero, Stampa Alternativa/Nuovi Equilibri, 2003 e 2004).

cento sulla massiccia occupazione di memoria che, in rapporto alle funzionalità fornite, non è in realtà così spropositata. La vendita di nastri Emacs fu la prima fonte di introiti per Stallman, consentendogli di dar vita a un'organizzazione non-profit dedicata alla gestione del progetto GNU: la Free Software Foundation (FSF), che alla prova dei fatti è forse un risultato anche più solido del progetto GNU. Il termine “software libero” è uno dei cavalli di battaglia di Stallman, secondo il quale la libertà è uno degli elementi fondamentali di software socialmente accettabile: libertà di utilizzo, di adattamento, di impiego anche in termini commerciali.

La FSF, fondata nel 1985, iniziò a porsi come punto di riferimento simile a quello che sarebbe divenuto pochi anni dopo il CSRG con il rilascio delle distribuzioni Net/1 e Net/2. Nei primi anni di lavoro, vennero così ricreati e resi liberamente disponibili i principali programmi di base caratteristici dell'ambiente Unix. Ovviamente, in assenza di una piattaforma di riferimento e di un kernel GNU, i programmi usavano le varie piattaforme Unix commerciali, sulle quali girava già software analogo. Ma i prodotti GNU, oltre a funzionare in maniera identica su una pletora di piattaforme hardware diverse, dimostravano sempre più spesso di avere maggiore elasticità e più funzionalità degli equivalenti proprietari, sicché anche amministratori di sistema non particolarmente idealisti si trovavano ad installarli per facilitare il proprio lavoro. Nel frattempo Stallman si dedicava a quelli che dal punto di vista tecnico sono forse i maggiori risultati del progetto GNU: il compilatore GCC e il debugger GDB.

Un compilatore è un programma che traduce le istruzioni da un linguaggio di alto livello al “linguaggio macchina”, cioè a

sequenze di bit direttamente eseguibili da un processore. Un debugger è invece un programma che consente di analizzare lo stato di un altro programma in esecuzione, interrompendolo, eseguendolo passo per passo e analizzando il contenuto della memoria. La scrittura di un compilatore e di un debugger è una delle “arti magiche” della programmazione, accessibile solo ai programmatori di prima classe. D’altra parte, questi strumenti sono il presupposto per la maggior parte dello sviluppo successivo su un sistema. GCC e GDB costituiscono oggi due progetti aperti tra i più complessi, e il loro aggiornamento è affidato a nutrite squadre di programmatori. Le versioni iniziali sono state scritte dal solo Stallman e hanno costituito i presupposti indispensabili per la crescita di GNU prima e più tardi per la nascita di Linux.

Un altro risultato fondamentale dell’attività di Stallman è sicuramente la GNU General Public License (GPL). Dopo l’esperienza fallimentare della comunità del laboratorio AI, Stallman era sufficientemente sfiduciato da capire che l’unico modo per garantire a un software libero di rimanere tale consisteva nella messa a punto di regole chiare e precise. Con una geniale applicazione “alla rovescia” delle leggi sul copyright, a volte indicata con il termine “copyleft”, a ciascun programma GNU venne applicata una licenza che garantiva a chiunque il diritto di utilizzarlo, copiarlo, modificarlo e ridistribuirlo, vietando però di aggiungere nuove restrizioni all’uso. In particolare, quando se ne distribuiva solo una versione eseguibile, andava garantita al destinatario la possibilità di accedere al codice sorgente e ogni versione modificata doveva continuare a essere coperta dalla GPL. Il primo tentativo di licenza in stile copyleft fu associato alla distribuzione di Emacs nel 1985, ma la prima versione ufficiale, la

GPL 1.0, venne utilizzata nel 1989. Dopo la versione 2.0 del 1991, il testo della licenza non ha subito ulteriori modifiche. Le clausole della GPL sono apparentemente semplici, ma le conseguenze risultano estremamente complesse da comprendere appieno. La stesura integrale occupa diverse pagine e, a uno studio approfondito, desta meraviglia l'attenzione con cui sono state considerate tutte le possibili eventualità. Per esempio, l'obbligo di applicare la GPL alle versioni modificate non sussiste se queste non vengono distribuite; in tal modo, il software GPL può essere adattato alle necessità specifiche di qualsiasi entità, senza l'obbligo di dover comunque procedere a una redistribuzione. Altro esempio: sebbene non sia possibile limitare la redistribuzione, è possibile richiedere una tariffa per ciascuna copia che si ordina e un'ulteriore tariffa per eventuali garanzie addizionali che si offrano a corredo. Questa la chiave di volta che ha consentito il fiorire delle versioni commerciali di Linux.

Quale sia lo stato attuale del progetto GNU è questione di opinioni. Secondo Stallman e i suoi seguaci, la maggior parte delle distribuzioni Linux in circolazione non sono altro che derivazioni del progetto GNU, sia pure non fedeli agli ideali originari: trattasi perciò di una realtà viva e in evoluzione. Secondo i detrattori, invece, il progetto GNU non ha mai raggiunto l'obiettivo primario, ovvero quello di realizzare un sistema operativo completo. Ciò è dovuto principalmente all'estrema complessità di HURD, il kernel ideato per GNU, a tutt'oggi in fase sperimentale. Resta comunque il fatto che i risultati di Stallman e della FSF, sia in termini tecnici che sociali, costituiscono le fondamenta su cui poggiano gran parte degli attuali sistemi open source.

Unix per tutti, tutti per Linux

Nei paragrafi precedenti abbiamo visto come la storia di Unix tra gli anni Settanta e Ottanta fosse stata un progressivo riadattamento del sistema verso piattaforme hardware sempre più leggere ed economiche. Malgrado negli anni Ottanta fossero stati introdotti sul mercato i primi personal computer (PC), la comunità Unix non li aveva presi in minima considerazione, ritenendo che fossero scarsamente potenti per l'utilizzo di Unix. Molti li ritenevano una piattaforma senza grande futuro e non vi dedicavano troppa attenzione. Fin dal 1980 Microsoft e SCO avevano creato Xenix, uno Unix per processori 8086, ma il sistema non ebbe grande diffusione e prese a sparire del tutto non appena IBM iniziò a promuovere i PC basati su DOS.

Una versione destinata ad avere vita più lunga, maggior successo e grandi ripercussioni, fu Minix, scritto da Andrew Tanenbaum, professore dell'Università di Amsterdam intorno alla metà degli anni Ottanta. Minix aveva fini puramente didattici. In effetti, AT&T aveva proibito di utilizzare il codice sorgente di Unix nei corsi universitari fin dal 1979, in una delle prime evidenti mosse verso la chiusura del sistema. Negli anni Settanta, Unix era diventato estremamente popolare anche per l'insegnamento nei corsi universitari, grazie soprattutto all'operato del professore australiano John Lions, i cui libri illustravano la struttura dei sistemi operativi commentando il codice sorgente di Unix. I "Lions", come sono noti i due volumi in ambiente Unix, sono ormai introvabili, e le uniche copie in circolazione vengono gelosamente custodite da ex studenti dell'epoca.

Negli anni Ottanta la maggior parte del mondo accademico si limitava quindi a considerare Unix da un punto di vista teorico. Tanenbaum optò invece per un approccio più pragmatico, scri-

vendo da zero un clone di Unix che non utilizzasse alcun codice proveniente da AT&T. Sembra si tratti del secondo clone completo mai scritto, essendo stato il primo un evanescente “Idris”. Ancora oggi Minix è probabilmente il miglior esempio di sistema operativo che sia possibile studiare in maniera completa, e uno dei pochi Unix in grado di girare su un processore 8086 con soli 640 KB di RAM. Il libro di Tanenbaum sui sistemi operativi, che costituisce la fonte di distribuzione principale di Minix, è entrato a far parte del novero dei classici dell’informatica.

Il 5 gennaio 1991 Linus Torvalds, sconosciuto studente dell’Università di Helsinki, se ne tornava a casa con il primo personal computer: un 386 a 33 MHz con 4 MB di RAM, uno dei migliori PC disponibili all’epoca. Ovviamente vi installò subito Minix. Cioè, non proprio subito: dovette attendere più di un mese l’arrivo della copia richiesta via posta ordinaria (unico modo per ottenerlo legalmente), a cui applicò le modifiche disponibili su Internet onde migliorarne le caratteristiche. Ciononostante, l’usabilità del sistema rimaneva ancora ben lontana dall’ideale; in particolare, l’emulazione di terminale, utile per connettersi da casa ai computer dell’Università, lasciava molto a desiderare. Torvalds decise così di avviare un progetto personale relativo all’emulatore di terminale. Ma invece di un programma per Minix, optò per un programma capace di interagire direttamente con l’hardware, cioè avviabile da dischetto. Una decisione progettuale decisamente originale, che per Linus significava sostanzialmente imparare il funzionamento del PC a livello hardware, ma che avrebbe cambiato il corso della storia.

Rapidamente Torvalds si ritrovò a utilizzare la posta elettronica e a partecipare ai newsgroup di Internet con il suo emulatore di terminale tramite i computer dell’Università. A quel punto sor-

se l'esigenza di scaricare i file e, dopo qualche esitazione, eccolo mettere a punto un driver per scrivere e leggere file dal disco rigido, dove risiedeva il file system di Minix, chiaramente documentato. Con neppure troppa attenzione, il programma andava trasmutandosi in un vero e proprio sistema operativo.

Così, ad inizio luglio, gli utenti Minix che leggevano il newsgroup *comp.os.minix* si videro recapitare un messaggio che solo un ingenuo poteva fraintendere:

[...]

From: torvalds@klaava.Helsinki.Fi (Linus Benedict Torvalds)

Per un progetto cui sto lavorando in minix, mi interesserebbe la definizione dello standard posix. Qualcuno sa indicarmi dove trovare, possibilmente in forma elettronica, la versione più recente delle regole posix? Un sito FTP sarebbe perfetto.

[...]

Non poteva certo sfuggire a quel newsgroup, composto in massima parte da esperti di Unix, che l'unico tipo di progetto per il quale fosse necessario lo standard Posix era la scrittura di un sistema operativo. La ricerca di Torvalds non ebbe esito, per ovvi motivi: gli standard dell'IEEE, come quelli di numerosi altri gruppi, vengono pubblicati per la vendita a un costo abbastanza elevato, per cui normalmente non circolano su Internet. Torvalds dovette quindi ricorrere al "piano B", ovvero riscrivere le funzionalità capaci di emulare quelle descritte nel manuale di una workstation Sun dell'Università di Helsinki. Procedura lunga e noiosa, ma certamente più veloce di quella che avrebbe comportato seguire passo passo uno standard della complessità del Posix.

A fine agosto, cioè a sette mesi dall'installazione di Minix, Torvalds poteva permettersi di scrivere, sempre via e-mail:

Saluti a tutti gli utenti di minix.

Sto scrivendo un sistema operativo libero (solo un hobby, niente di grande e professionale come gnu [Hurd]) per cloni AT 386(486). Ho iniziato ad aprile, e comincia ad essere pronto. Mi piacerebbe avere indicazioni sulle cose che piacciono/non piacciono in minix, dal momento che il mio SO gli assomiglia vagamente – ad esempio, nella disposizione fisica del filesystem (per motivi pratici).

[...]

Il messaggio suscitò un pugno di repliche entusiaste e nacque- ro i primi contatti con altri sviluppatori interessati a collaborare. Dopo di che molto silenzio: la versione 0.01 di quello che già si chiamava Linux fu pubblicata il 17 settembre 1991, ma il primo annuncio pubblico, sempre su *comp.os.minix*, ormai celeberrimo, è datato 5 ottobre:

Rimpiangi i giorni felici di minix-1.1, quando gli uomini erano uomini e scrivevano i propri driver? Non hai un progetto carino e muori dalla voglia di sfogarti su un SO che puoi provare a modificare a volontà? Inizi a trovare frustrante quando lavorando con minix ti funziona tutto? Niente più sessioni di lavoro notturne per far funzionare un bel programma? Questo annuncio potrebbe fare al caso tuo :)

[...]

I codici sorgenti di questo mio progettino si trovano su

nic.funet.fi nella directory /pub/OS/Linux. La directory contiene anche qualche file README e un paio di eseguibili per lavorare con Linux (bash, update e gcc, che altro si può desiderare? :-). L'intero codice sorgente del kernel è disponibile, dal momento che non è stato utilizzato codice minix. I codici sorgenti delle librerie sono solo parzialmente liberi, per cui non possono essere ancora distribuiti. Il sistema può essere compilato così com'è e funziona. I codici sorgenti degli eseguibili (bash e gcc) si trovano nello stesso posto, nella directory /pub/gnu.

Come racconta lo stesso Torvalds nel libro-intervista *Just for fun*², il vero salto di qualità di Linux avvenne con l'introduzione del *page-to-disk*, la caratteristica del kernel che permette di salvare parte della RAM su disco e quindi utilizzare programmi che non entrerebbero nella memoria disponibile. Minix era privo di una simile caratteristica, molto importante invece a livello pratico: Linux iniziava ad entrare nel novero dei sistemi "professionali". E all'inizio del 1992 gli utilizzatori raggiungevano già le centinaia. Così, arrivato alla versione 0.12, Linus cancellò la nota di copyright molto semplice che aveva utilizzato all'inizio per sostituirla con la GNU GPL. Eccoci così di fronte a un altro evento assai importante nella storia di Linux perché, all'interno della comunità Unix, il significato della licenza era già ben chiaro. Una frase del tipo "Linux è un kernel per piattaforma AT 386/486, con un file system minix e funzionalità di *page-to-disk*, scritto da zero e rilasciato con licen-

² *In italiano: Rivoluzionario per caso. Come ho creato Linux (solo per divertirmi), Garzanti Libri, 2001. Per ulteriori dettagli sulle vicende legate all'avvento di Linux e dell'open source si veda inoltre: Codice ribelle. La vera storia di Linux e della rivoluzione Open Source, Hops Tecniche Nuove, 2002.*

za GPL” era sufficiente a far rizzare le orecchie a un hacker pro-
vetto. In pratica: il 386 era il primo processore con capacità tali
da far pensare a un utilizzo di un sistema di tipo Unix; i sistemi
BSD erano impelagati nella causa con AT&T; del kernel GNU
non si vedeva traccia; Minix era bloccato dalla licenza di Tanen-
baum, che voleva mantenerlo un sistema semplice e chiaro da
utilizzare per l’insegnamento; e i pochi Unix commerciali non
erano interessanti per i potenziali utenti perché non modificabi-
li a livello di codice sorgente. Grazie a questa serie di motivi coin-
cidenti, la popolarità di Linux esplose con una rapidità impres-
sionante. Nella primavera del 1992 Linux poteva già utilizzare il
sistema grafico X Window. Il passo successivo era aggiungere le
capacità di comunicazione in rete, cioè l’equivalente del codice
Net/1 di BSD. Anche questo venne riscritto da zero e ci vollero
quasi due anni. La versione 1.0 di Linux venne annunciata nel
marzo del ‘94 ed era ormai un sistema noto e popolare.

L’apertura delle sorgenti

Nel 1997, Linux era ormai un fenomeno, sia pure limitato
all’ambito accademico e al novero degli appassionati. Era comun-
que più che abbastanza perché si tenessero veri e propri conve-
gni dedicati al nuovo sistema. Al *Linux Kongress* di Würzburg,
Eric S. Raymond, che abbiamo già incontrato come autore del-
la parodia “Unix Wars”, presentava un saggio che sarebbe dive-
nuto celeberrimo, *La Cattedrale e il Bazar*, in cui si analizzava-
no, in termini estremamente pragmatici, il modello di sviluppo
di Linux e le ragioni del suo successo. L’articolo di Raymond
suscitò subito grande interesse, pur se con le inevitabili critiche,
ed ebbe ricadute estremamente importanti per il suo autore.
In quegli anni, la Netscape Corporation, che aveva risvegliato il

mondo business al fenomeno del World Wide Web, iniziava a perdere la battaglia per il dominio del mercato browser, sotto l'incalzare di Internet Explorer. Netscape aveva reso liberamente utilizzabile il programma, per concentrarsi così sullo sviluppo delle tecnologie lato server e quindi lavorare con le aziende interessate ad una presenza sul Web. La disponibilità di Netscape Navigator garantiva che chiunque poteva semplicemente installare un nuovo programma e accedere alle pagine web costruite con tecnologia Netscape. La storia di come Microsoft sia riuscita a scalzare il predominio di Netscape e diffondere l'uso di Internet Explorer è troppo lunga per essere riportata in questa sede. È sufficiente ricordare che Netscape stava considerando la diffusione del proprio browser come prodotto libero, sotto una licenza in stile GPL. L'articolo di Raymond ebbe un impatto sufficiente a farlo convocare dai dirigenti di Netscape onde pianificare la strategia di rilascio pubblico dei codici di Navigator. Il frutto di quella strategia sono la Mozilla Foundation e l'annessa linea di programmi open source, sebbene solo di recente l'usabilità dell'omonimo browser sia divenuta tale da permettergli di competere a tutti gli effetti con Internet Explorer.

Il rilascio pubblico dei codici di Netscape Navigator venne annunciato verso la fine del gennaio 1998. Qualche settimana dopo, in quel di Palo Alto, in California, si svolse una riunione tra i vari esponenti della comunità che gravitavano intorno a Linux e ai programmi liberi per discutere sull'annuncio pubblico relativo a Netscape. Era la prima volta che un'azienda commerciale decideva di rilasciare liberamente uno dei prodotti di punta, per cui i tempi sembravano maturi per un'adozione più ampia dei modelli di sviluppo aperti nell'industria del software. Uno dei problemi che vennero affrontati fu la man-

canza di una dizione adeguata che caratterizzasse i programmi i cui codici sorgenti fossero liberamente disponibili, modificabili e distribuibili. L'unica espressione comunemente condivisa era quella utilizzata nell'ambito del progetto GNU, "software libero" – ma questa veniva considerata fortemente appesantita dall'attitudine intransigente di Stallman. Inoltre, nell'originale inglese, è presente l'ambiguità per cui "free" significa non solo "libero", ma anche "gratuito": un elemento che in molti casi poteva risultare ambiguo e poco gradito nel mondo commerciale. Così, nel corso della riunione, fu coniato un nuovo termine per identificare sia i programmi utilizzati sia la sottostante impronta ideale: "open source" (traducibile in italiano solo come "codici sorgenti aperti", che è abbastanza lunga e incomprensibile da far prediligere il mantenimento della dizione inglese).

Nei giorni successivi, numerose figure di rilievo del settore approvarono la scelta del nuovo termine. Immediatamente vennero create linee guida per definire quali fossero le condizioni essenziali perché la licenza associata a un programma potesse definirsi "open source": sono una decina di condizioni piuttosto lunghe da analizzare e spiegare in dettaglio, ma quelle di base sono fondamentalmente la libertà di distribuzione, l'accesso al codice sorgente e la possibilità di ricavare lavori derivati. Il sito della "Open Source Initiative", che è divenuta un'organizzazione non-profit, mantiene un elenco delle licenze ufficialmente approvate. Ad aprile 1998, vi fu anche una votazione ufficiosa a un convegno organizzato da Tim O'Reilly, un editore di libri tecnici che dedica particolare attenzione ai programmi liberi. Nel corso del '98, l'espressione si diffuse a macchia d'olio in tutta la comunità. Lo stesso Stallman prese in considerazione l'idea di utilizzarla, ma alla fine decise che fos-

se più importante mantenere vivo il riferimento alla “libertà”. A tutt’oggi, la dicotomia software libero-open source è probabilmente la più grande divisione ideale che caratterizza quella che può a tutti gli effetti essere considerata un’unica, grande comunità. Ed è una grande sorgente di confusione, perché molti, pur di mostrarsi politicamente corretti, utilizzano contemporaneamente entrambe le espressioni. Nonostante tutto, l’espressione “open source” è ormai così ampiamente diffusa che a livello pratico comprende anche il progetto GNU e la FSF; sebbene, ovviamente, Stallman e i più stretti seguaci non siano d’accordo con questa visione.

Come auspicato nella riunione di Palo Alto, la Netscape è stata solo la prima di una lunga serie di aziende che decisero di rilasciare il proprio software con una licenza di tipo open source, oppure di adattarlo per poter girare in ambiente Linux. In tal senso, un altro programma dal notevole impatto è stato OpenOffice.org, reso disponibile da Sun Microsystems, e che costituisce il primo caso di suite per ufficio con licenza open source. Tuttavia, l’analoga apertura delle sorgenti avviata da IBM con l’ambiente di sviluppo Eclipse, presentava forse costi operativi maggiorati, oltre a un’utenza base assai più limitata. Ecco perché a tutt’oggi, il supporto più entusiasta all’open source in generale, e a Linux in particolare, viene proprio da IBM. Il gigante dell’elettronica, dopo il fallimento del proprio sistema operativo desktop, OS/2, vede probabilmente in Linux l’arma migliore per tentare di arginare l’invasione da parte di Microsoft di tutti i settori del software e delle comunicazioni elettroniche. E nell’ambito della comunità open source è generalmente considerato il più importante alleato commerciale.

In principio era Novell...

Nel 1993 Novell aveva tutto: un florido mercato nell'ambito dei server, uno dei manager più duri dell'universo, una solida tradizione tecnologica e un laboratorio di sviluppo che sfornava pezzi di ottima tecnologia al solo schiocco delle dita dei propri ingegneri. Era la seconda azienda informatica del mondo e tutti si sarebbero attesi un fulgido futuro e una perenne posizione di testa nel firmamento delle stelle di prima grandezza del panorama dell'information technology.

Ma non andò così.

Nata a Provo nello Utah per mano di Jack Davis e George Canova nel 1979, la Novell Data System aveva iniziato come semplice costruttore di terminali per mainframe e successivamente per microcomputer e sistemi operativi su disco, lanciandosi nel fiorente business dei primi sistemi personali sulla scia del successo di Altair e di Apple.

Quando nel 1983 venne rilevata da Raymond J. Noorda, grazie agli investimenti della *Safeguard Scientific*, dopo un incontro casuale con il management in un hotel di passaggio, l'azienda di Provo era collassata da 120 a soli 15 dipendenti e non sarebbe sopravvissuta un ulteriore inverno. Mentre Apple, Commodore, Atari e Microsoft si spintonavano nel mercato dell'informatica per tutti, Noorda, un rigido mormone, tentò di costruirsi in casa un piccolo ordigno esplosivo per dare vita alle nuove idee di capitalismo moderno intriso di richiami morali.

Raymond Noorda, ingegnere, capitalista

Noorda, ingegnere d'esperienza e capitalista per censo, curatore dell'ingente fortuna di famiglia raccolta nella società d'investimento NFT, Noorda Family Trust, si pose a capo di Novell e la trascinò in uno strano miscuglio di competizione senza quartiere e cooperazione leale, dando smalto alle idee visionarie dei veri cervelli tecnologici dell'azienda, Craig Burton e la moglie Judith Clarke, messe poi in pratica dall'ottimo tecnico Drew Major.

Alla fine degli anni Ottanta il mondo dell'alta tecnologia parlava il linguaggio di Novell, che aveva applicato – ben prima della stessa Microsoft – la pratica di “embrace & extend”, ovvero l'uso di tecnologie non inventate in modo autonomo ma migliorate ed estese per renderle più avanzate e più aperte, più disponibili e semplici. In seguito il declino inizierà proprio quando, con la scomparsa dei Burton, l'azienda mostrò di aver dimenticato completamente le proprie origini e, peggio ancora, iniziò a negare la stessa importanza di quell'apertura architeturale che aveva fatto di NetWare il principale prodotto dell'era del networking.

Interessato agli aspetti più teorici del mercato tecnologico avanzato, Noorda, capitalista atipico con una forte anima morale, usò Novell per teorizzare un modo di fare business completamente nuovo per l'America. Al sacro fuoco della competizione a tutti i costi che animava i diretti concorrenti, Noorda contrappose una invenzione tutta propria, la *co-ompetizione*, un inedito miscuglio tra cooperazione e competizione: per la prima volta venivano prese in considerazione quelle catene di valore che si creano in una rete aperta dove le aziende, pur competendo su un mercato comune, si sostengono a vicenda laddo-

ve possono cooperare, ad esempio nella creazione di standard comuni e di funzionalità condivise. Noorda dimostrò di essere un navigatore molto capace nel gioco delle alleanze competitive, anticipando di 20 anni le teorie dell'impresa di rete tanto di moda nella New Economy d'inizio millennio.

La piccola Novell esplose letteralmente, abbandonando la realizzazione di 'semplici dispositivi di connessione remota ai mainframe' per lanciarsi nel fantascientifico campo delle reti locali. Un campo, così avanzato eppure così vicino, in cui nel breve volgere di tre anni Novell sancì completamente la supremazia del proprio software: NetWare. L'azienda di Provo vendeva le schede di rete Ethernet, oggi comuni in qualsiasi PC, quasi sotto costo pur di abilitare il lavoro in rete.

E Noorda non commise neppure gli errori classici delle aziende high-tech di allora. Vendeva solo quello che poteva effettivamente realizzare e realizzava solo quello che poteva effettivamente vendere. Attentissimo allo spirito di comunità che si respirava nell'ambito del boom informatico, Novell dette inoltre vita ad una grande quantità di iniziative collaterali, tra cui riviste come *LAN Times*, trade-show tecnologici quali *NetWorld* e un canale di rivenditori e partner professionali che raramente ha avuto un equivalente fino ai nostri giorni attuali, tipo il *Certified Network Engineer*. E, com'è ben noto ancora oggi, bastava dire Rete o LAN per intendere Novell.

Novell Inc. divenne insomma il prototipo di 'azienda-rete', dove il concetto di network assumeva valore sia sistemico ed organizzativo che operativo e produttivo. Negli anni Ottanta, le reti Novell dilagavano e la struttura commerciale non aveva eguali. Viveva in rapporto simbiotico con la tecnologia stessa dell'informatica, proliferava dovunque fosse possibile trovare

uno slot libero in cui inserire una scheda di rete Ethernet e dovunque esistesse un minimo di memoria per caricare un driver NetWare: i produttori di computer si avvantaggiavano dell'esistenza della rete e Novell era la rete. Mentre tutti gli altri produttori si baloccavano con le proprie soluzioni proprietarie e pasticciate – IBM con TokenRing, Digital con DECnet e persino Microsoft tentò una MSNet – Novell abbracciò fieramente gli standard aperti di Xerox, Ethernet e XNS per estenderli in un complesso di hardware e software senza eguali.

Il confine di Novell era quindi limitato solo dalle dimensioni del mercato, non ancora così vasto da renderla del tutto vittoriosa. Nel senso che una simile rete locale non riuscì ad imporsi come “killer application”, uno di quei servizi tali da convincere gli utenti a superare il notevole costo iniziale necessario ad installare un sistema computerizzato tale da godere dei servizi (relativamente a buon mercato) di Novell. E aveva un altro limite, quella rete: funzionava. Software fatto bene, solido e ben testato. Senza fronzoli inutili e senza concessioni alle mode, il sistema di rete si installava al termine di una procedura lineare e senza dubbi, aveva limitate possibilità di adattamento e praticamente si manteneva da sola. I contratti di manutenzione diventavano quasi orpelli inutili sul prezzo dell'installazione.

L'inizio della fine

Nel 1993 Microsoft contattò Noorda due volte per convincerlo a vendere Novell. Nel 1993, infatti, Novell e Microsoft non erano in diretta competizione, quanto piuttosto complementari. La prima vantava il monopolio delle reti locali, la seconda quello delle interfacce utente.

Il leader di Novell l'aveva rilevata a 58 anni e per i successivi 12

si era comportato da padre-padrone, giorno per giorno in azienda, occupandosi anche del più insignificante dei dettagli. Un paio di by-pass cardiaci ne mettevano a repentaglio quasi quotidianamente lo stato di salute. Noorda aveva già annunciato che, in occasione del suo settantesimo compleanno, si sarebbe ritirato a vita privata. Novell pareva dunque destinata ad una sorte invidiabile: confluire nel più potente gruppo informatico del mondo.

Ma Bill Gates usò una delle strategie più machiavelliche: ventilò l'acquisizione, studiò ben bene l'azienda e poi si ritirò, mollando tutto senza neppure dare spiegazioni. Ci sarebbe voluto molto di meno per far imbestialire chiunque. Ancor peggio, Noorda si rese conto che si trattava dell'inizio della fine.

Quando, nell'aprile del 1994, compiuti 70 anni, si ritirò dalla gestione diretta dell'azienda e nel dicembre dello stesso anno abbandonò anche il consiglio di amministrazione per andare in pensione, Novell occupava ancora un posto nel firmamento delle stelle info-tech. Da allora iniziò tuttavia un anno orribile per l'azienda di Provo.

La transizione tra Noorda e Frankenberg, il manager esperto proveniente da Hewlett-Packard che avrebbe dovuto sostituirlo, si palesò subito come più complicata di quanto previsto. Tipo debole e supponente, Frankenberg era interessato alla propria carriera assai più che all'azienda. Partì subito col piede sbagliato, trattando in gran segreto l'acquisizione del software Money solo per togliere la patata bollente dalle mani di Microsoft, in grossa sofferenza a causa delle indagini che l'Antitrust USA aveva avviato nel settore del software per la gestione personale delle finanze.

Non avrebbe potuto compiere mossa peggiore: aiutare il nemi-

co. L'azienda gli si rivoltò contro, anche perché l'ultimo anno di Noorda in Novell era stato un turbinio di acquisizioni che molti avevano giudicato stupide prima che assurde, pur se Ray aveva agito sulla base di informazioni di cui altri non potevano disporre. Fu un'affannosa rincorsa per la realizzazione di una linea Maginot virtuale contro Microsoft, la cui proposta d'acquisto poteva voler dire solo una cosa: Microsoft aveva intenzione di scendere nel campo dei NOS ("Network Operating Systems", ovvero "sistemi operativi di rete", i quali permettono la condivisione di risorse di calcolo nell'ambito di una rete informatica), anche se s'affannava a dire il contrario. E lo avrebbe fatto con una ingente quantità di denaro fresco, sottratto agli utili degli altissimi margini di guadagno su MS-DOS.

I muscoli della tecnica

Noorda aveva le idee chiare su cosa fare quand'era in sella a Novell e viveva lo scontro nel mercato come una sorta di prova muscolare di virilità: non solo non permetteva che qualcuno tentasse di intimidirlo, ma era sempre il primo a ricorrere alle maniere forti, qualunque cosa ciò volesse dire.

Quando a fine 1992 aveva acquisito, da una AT&T esangue, gli Unix Software Labs (USL), chiuse l'annosa causa contro l'Università della California e BSDI (Berkeley Software Design Incorporated), sostenendo che aveva ogni intenzione di competere sul mercato. Tutti pensarono che stesse dando una lezione allo stracco management AT&T, il quale aveva trascinato per anni una causa legale contro i mulini a vento senza ottenere alcunché di significativo. Invece l'obiettivo reale era Gates, l'ex-ragazzino di Seattle, che pure con Unix aveva flirtato per quasi 10 anni realizzando il Microsoft XENIX.

Con Unix tra le mani, Noorda si sentiva più forte, ma quando le acquisizioni presero di mira le tecnologie ormai decotte della DRI (la ex Intergalactic Digital Research Incorporated del leggendario Gary Kildall, morto l'anno precedente per uno strano scivolone all'uscita di un ristorante), produttrice del CP/M - primo vero sistema operativo per microcomputer - e successivamente del DR-DOS (Digital Research Disk Operating System), nonché del boccheggianti programma di videoscrittura WordPerfect e, in ultimo, dello zoppicante foglio elettronico Quattro Pro, l'obiettivo finale divenne chiaro a tutti: la competizione globale con Microsoft.

Ma altrettanto chiaramente si diffuse la sensazione che Novell non avrebbe mai potuto farcela.

WordPerfect era un prodotto qualitativamente superiore quando Microsoft ebbe l'ardire di rilasciare pubblicamente Word 1.0, pieno di bachi e funzionante a stento su DOS. Ma Bill Gates inondò il mercato di copie gratuite, allegando per la prima volta un supporto magnetico ad una rivista informatica di ampia diffusione quale *PC World*. Da un giorno all'altro, decine di migliaia di copie di un word-processor furono disponibili per gli utenti, un'azione di dumping senza precedenti nella storia del mercato statunitense. Nei dieci anni successivi WordPerfect, pur mantenendo un buon nome come prodotto di qualità, vide il proprio mercato quasi azzerato.

Fu un brutto colpo anche per Ray Noorda, che pure se ne avvantaggiò dieci anni dopo, seppur per semplice campanilismo. Anche la WordPerfect Corporation era dello Utah e Noorda vi aveva probabilmente investito denaro dalla finanziaria di famiglia. Nel 1994, WordPerfect rappresentava sicuramente un software di buon livello, pur rimanendo ampiamente sotto lo

standard di Word – anche perché quest’ultimo poteva attingere, in fase di sviluppo, alle infinite risorse che Windows garantiva a Microsoft e non aveva bisogno di guadagnarne direttamente onde sostenersi. L’unione tra Novell e WordPerfect fu agevolata anche dall’interesse di Lotus, già produttrice di una suite da ufficio che aveva proprio nel word-processor un punto debole. Ma più di questo contava il fatto che la combinazione Lotus più WordPerfect avrebbe potuto scalzare la stessa Novell dal secondo posto in classifica tra le software-house statunitensi.

Gli altri acquisti della Novell assunsero invece poco più di un valore storico: il CP/M era ormai in pensione da almeno 10 anni e il DR-DOS, pur validissima alternativa all’ultima versione di MS-DOS, scontava chiaramente il non trascurabile problema che Microsoft poteva rallentarne il funzionamento sotto Windows, ormai divenuto l’interfaccia d’elezione per i PC moderni. Microsoft invece si limitò ad impedire direttamente l’installazione del DR-DOS, dopo aver lanciato un’incredibile campagna denigratoria.

Paradossalmente, però, fu la stessa Novell a dare il colpo di grazia al proprio DOS. Infatti quando, dopo innumerevoli ritardi, mise in campo l’ultima versione di DR-DOS (rinominato Novell DOS 7.0), la qualità del programma appariva notevolmente peggiorata rispetto all’ultima versione di Kildall, la velocità drasticamente ridimensionata e nel complesso il prodotto dava l’impressione di non possedere alcun valore d’uso. Insomma, un vero flop commerciale senza precedenti nella storia dell’azienda.

Nell’arco di pochi mesi, Novell DOS 7.0 venne così rifilato, a prezzi di saldo, ad una piccola azienda locale di Lindon in Utah,

la prima delle numerose Caldera che incontreremo in questa vicenda.

Esaurite le acquisizioni balorde, Noorda finì in pensione e ognuna di quelle tecnologie prese la propria strada fuori dal business di Novell: mentre DR-DOS passò a Caldera nel 1996, WordPerfect e Quattro Pro vennero venduti a prezzi di liquidazione a Corel, altra combattiva ditta anti-Microsoft di proprietà del canadese Michel Cowpland che rimarrà per anni spina nel fianco di Bill Gates, fino a venirne quasi fagocitata nel 2001.

Lo UNIX che era stato acquisito dagli USL fu rilasciato per la prima volta sul mercato il 2 novembre del 1992, come Novell UnixWare 1, così da riecheggiare almeno nel nome il vecchio NetWare. Ma il prodotto non ebbe il successo sperato e il nuovo management Novell decise di disfarsene. Questo fu il secondo regalo di Frankenberg a Microsoft, anche se le pressioni della comunità dei produttori e degli sviluppatori Unix lo costrinsero, letteralmente, a suddividere i diritti acquisiti da AT&T: il marchio e tutti i diritti connessi alla definizione di Unix vennero rilasciati alla X/OPEN, consorzio di produttori di sistemi Unix basati o meno su codice AT&T, mentre Santa Cruz Operations acquisì il codice originale AT&T (quello che era stato ottenuto nel 1989 dalla fusione di XENIX, Sun OS, BSD e System V, chiamato allora System V Release 4 o SVR4).

La fine dell'età dell'oro

Il pensionamento di Ray Noorda lasciò Novell nei guai, con un manager quanto mai ambiguo e la prospettiva di uno scontro sanguinoso con Microsoft. Ma lasciò anche l'anziano finanziere dello Utah con molto tempo libero, influenti conoscenze politiche e parecchi soldi a disposizione.

Noorda era ormai noto per l'odio viscerale nei confronti di Bill Gates, al quale IBM aveva fatto il più bel regalo che ogni capitalista avrebbe mai chiesto: un mercato nuovo di zecca tutto da conquistare. Se fosse successo a lui, Mr. Noorda, non si sarebbe limitato a guadagnare solo così tanto. Avrebbe sbancato alla grande, e a quest'ora, chissà, magari avrebbe in mano le redini della maggiore azienda informatica del mondo. Grazie alle proprie amicizie politiche, forse avrebbe preteso qualcosa di più di una candidatura di bandiera per le presidenziali come quella di Ross Perot (già presidente dell'azienda informatica di casa General Motor, la EDS). Lui, sempre Ray Noorda, avrebbe addirittura rischiato di diventare Presidente degli Stati Uniti. Comunque sia, lo sviscerato odio per Microsoft gli era valso un titolo pubblico del tutto particolare, "la Ray Noorda Syndrome", coniata da Sheldon Laube (CEO di CenterBeam), come esegesi della retorica anti-Microsoft. E d'altronde Noorda aveva più di una ragione per prendersela con Gates. Non solo il malefico Bill gli era andato a pestare i piedi nella bomboniera dei sistemi di rete da lui inventata, riuscendo ad imporre i propri prodotti al mondo intero, ma ne aveva infranto anche le ambizioni politiche e, quel che è peggio, lo aveva banalmente buggerato con la falsa storia dell'acquisizione onde carpirgli i migliori segreti aziendali. Il ragazzino di Redmond aveva sbeffeggiato l'anziano mormone che si sentiva l'incarnazione stessa della "cattiveria manageriale americana" (utilizzata a fin di bene, ovviamente, e non c'è miglior bene che la creazione del profitto).

Era notoria la sua predilezione per l'ambito politico e proprio all'eccellenza tecnologica di Novell e di WordPerfect tutti accreditavano una consistente fetta occupazionale di primissimo

livello nello Utah, a disposizione dei programmatori di mezzo mondo. Lo Utah era lo stato dell'Unione noto, ben prima di Noorda, più per la poligamia delle comunità dei mormoni che l'avevano fondata nel 1847 che per l'alta tecnologia — sebbene avesse dato i natali anche ad Alan Kay, uno dei più importanti ed influenti ingegneri dello Xerox PARC, e al giovane Nolan Bushnell, fondatore di Atari e padre dei videogame. Ma il 1993 doveva rivelarsi un anno orribile per Noorda, Novell e lo Utah. Microsoft, con una mossa certamente inattesa, decise di scendere nella competizione dei sistemi operativi di rete (NOS), lanciando il proprio Windows for Workgroup 3.11, una semplice estensione del DOS con una minima interfaccia grafica ricalcata su una brutta copia del Macintosh di Apple e con appiccicato alla bell'e meglio un sotto sistema di connessione di rete, pasticciato da vecchi standard IBM estesi a proprio uso e consumo. Il tutto in maniera abbastanza traballante e così poco professionale da impedire praticamente ogni vero utilizzo professionale.

Ma il segnale era fin troppo chiaro e Novell iniziò a tremare.

Il regista dell'Antitrust

Lapidario fu il giudizio dei posteri su quei giorni di Ray Noorda e di Novell, come riportò Jerry Pournelle sulle colonne di *Byte*, la più nota rivista del settore. Novell aveva mancato l'occasione per innovare la propria visione aziendale e i propri prodotti all'interno del mercato e della tecnologia correnti, perché focalizzato unicamente sull'odiato nemico. L'abilità propositiva di Noorda negli anni Ottanta era stata la fortuna di Novell. Ma negli anni Novanta, l'incapacità di sottrarsi al combattimento a tutti i costi, da parte dello stesso Noorda, ne decretò

la rovina. E quando l'azienda riuscì a tirarsi fuori dal suo influsso, ormai c'era poco da ricostruire.

Noorda però non poteva accettare di essere estromesso e di accreditare così la verità di essersi fatto fuorviare negli affari da un odio ingiustificabile. Assetato di vendetta, eccolo divenire la vera anima nera all'opera dietro il processo Antitrust aperto dal governo statunitense contro Microsoft.

Era passata alla storia la penosa vicenda dello spezzatino AT&T, obbligata a dividersi onde tutelare la concorrenza nel mercato nord-americano, con il risultato di aver bloccato il colosso telefonico nell'estendere la propria supremazia al resto del mondo e di aver spalancato le porte alla debacle tecnologica statunitense rispetto agli exploit europei di fine millennio. Dopo una tale vicenda, nessuno aveva davvero voglia di scalfire nuovamente un simbolo del toro economico a stelle e strisce che era già riuscito ad imporsi a livello mondiale.

In maniera più rapida e consistente di quanto fosse mai riuscita a fare qualsiasi altra blasonata azienda nazionale, più della storica Coca-Cola, più di Mc Donald's e di Nike, Microsoft era non solo riuscita ad estendere il proprio marchio a livello globale, ma era penetrata nella vita quotidiana della maggioranza delle realtà produttive del pianeta.

Mettere in dubbio la possibilità di Microsoft sul mercato nazionale significava mettere in dubbio le potenzialità degli Stati Uniti nel mercato tecnologico planetario. Il mondo delle ricette economiche semplici alla George Bush non aveva alcuna intenzione di imporre a Bill Gates qualsivoglia limite, avendogli peraltro conferito molto prematuramente nel 1992 il National Medal of Technology for Technological Achievement, riconoscendone "la prematura visione del computing universale in

tutte le case e gli uffici, le capacità tecniche e manageriali per aver creato un'azienda tecnologica mondiale, e il contributo allo sviluppo dell'industria informatica personale”.

Ma i tempi stavano cambiando. L'epopea mitica della famiglia Gates si sosteneva da sé sotto la fanfara filo-liberista del presidente Bush. I Gates, notoriamente e dichiaratamente repubblicani, non avevano avuto nessun bisogno di fare promozione politica alle proprie attività.

Microsoft, cresciuta troppo in fretta e senza alcun supporto dal governo federale, non aveva avuto nessun bisogno di stabilire una strategia complessa per stabilire un buon rapporto con la politica. Si può dire che fino ad allora il successo del piccolo Bill fosse sfuggito all'assetato mondo della raccolta-fondi elettorale, e che la sua indifferenza politica, rasentando l'ignavia, fosse stata un'ottima arma per evitare di schierarsi.

Ma non poteva essere più così nelle elezioni del 1992. In campo erano scesi Bill Clinton e Albert Arnold Gore Jr., figlio del senatore Albert Gore Sr., che fu il grande promotore della realizzazione della vasta rete autostradale statunitense. Onde emulare il maggiore successo politico del padre, Albert Gore Jr. lanciò un ardito parallelo tra le autostrade reali, che fecero la forza commerciale dell'America del dopoguerra, e le autostrade telematiche, che avrebbero dovuto realizzare l'eccellenza statunitense del futuro. Gore e Clinton, coetanei di Gates, erano lì a pestargli i piedi e non era proprio verosimile che trascurassero Microsoft. Situazione difficile per Gates, dunque, che dovette inventarsi praticamente dal nulla una capacità di fare lobbying a lui poco congeniale. Secondo un'inchiesta di Matt Loney della testata high-tech inglese *ZDNet UK*, l'impegno finanziario del Microsoft Corporation Political Action Committee, la struttura di

lobbying che a livello locale e federale si occupa di sostenere gli interessi di Microsoft, nel 1995 non superava i 16.000 dollari, con un solo dipendente a darsi da fare in quel di Washington. Ma nel ciclo elettorale del 2000 ecco raggiungere la bella somma di quasi 2.5 milioni di dollari, il 170 per cento di quanto, ad esempio, la Enron aveva distribuito durante la stessa campagna. Chi invece queste opportunità non se le faceva sfuggire era Ray Noorda, abituato com'era fin da giovane a trattare con le autorità governative del proprio stato, a barcamenarsi per ottenere sgravi fiscali e leggi di supporto alle proprie aziende, potendo altresì contare sull'immensa rete relazionale delle famiglie mormoni.

Il sogno tecnologico dello Utah, in cui Noorda e l'annesso network avevano fortemente creduto, e che era andato concretizzandosi per oltre dieci anni, portando prosperità e ricchezza, benessere economico e tranquillità, era stato infranto dalla prepotenza di Microsoft. Con quel modo certamente immorale di trattare, non tanto gli affari, quanto le persone – almeno agli occhi di Noorda. Non a caso il primo licenziato da una azienda informatica dello Utah fece così scalpore da sollevare una gara di solidarietà. Ma non fu certamente l'ultimo.

In ogni caso, quello che non era riuscito a realizzare dentro Novell, Ray Noorda decise di tentarlo al di fuori.

Intanto in Novell...

Gli anni orribili di Novell avrebbero depresso chiunque, i margini operativi diventavano ogni giorno più ristretti e lo spazio di manovra per l'azienda collassava. Ma, ancor peggio, il management interno sembrava candidarsi alla caricatura del capitano che affonda inerte al timone del proprio vascello.

Anche se nelle dimostrazioni animate, Novell si raffigurava come un veliero che cannoneggiava ed affondava Microsoft a colpi del proprio directory service NDS (“Novell Directory Services”, un sistema di indicizzazione delle risorse di rete simile al posteriore “Active Directory” di Windows 2000), in realtà nell’azienda si respirava ben altra aria. Quando Frankenberg fu indotto all’abbandono, si parlò di “difetto di esecuzione” degli ordini del capitano da parte dell’azienda. La verità era che il capitano aveva imposto una sterzata talmente repentina, con un immediato rivolgimento di fronte, che nessuno avrebbe potuto ragionevolmente stargli dietro.

Eppure i programmatori Novell, per quanto tutto intorno a loro sembrasse crollare, si trovavano in una posizione invidiabile. Avevano per le mani tutte le tecnologie dei sistemi operativi sviluppate nei precedenti venticinque anni e più. Da XNS della Xerox ad Ethernet, da CP/M al DOS, da NetWare ai moderni Directory Services, dai più astrusi protocolli proprietari di rete ai moderni standard Internet: tutto era loro pienamente accessibile. E tutto era coperto da un qualche diritto. Ma soprattutto Novell aveva Unix.

Pur se nessuno, tra il nuovo management Novell, ci credeva veramente, i tecnici scalpitavano. Unix, sostenevano, possedeva grandi potenzialità e non solo nel “core business” di Novell, ma in tutte quelle aree in cui il vecchio Noorda voleva competere con Microsoft. Come sistema operativo di rete, come desktop utente, come sistema di produttività individuale, come server Internet.

L’amorfo management Novell tentennava. L’aria era pesante. L’azienda tentò di riproporre i propri servizi di directory e si lanciò in una crociata pro Java a causa dell’attrazione nell’orbita Sun.

L'ambiente diventava sempre più opprimente per chi assisteva all'evoluzione di Unix tra le masse di utenti avanzati in Internet. Questo Unix, di nome Linux, stava travolgendo le dighe degli Unix proprietari, era dotato di programmi solidi come Sendmail per la posta elettronica, Samba per la realizzazione di servizi di disco e stampa remoti e Apache come server Web. Sembrava essere molto più di una promessa. Era ormai una realtà.

Novell era bloccata di nuovo nella competizione con Microsoft e tentava una serie di mosse per sottrarre mercato al nuovo nato di casa Microsoft: NT. Ma l'impresa appariva disperata. Pur se lento e inaffidabile, nonché basato su un kernel traballante, quest'ultimo era dotato di un'interfaccia utente a finestre derivata da quella di Windows. Veniva inoltre propagandato con grande dispendio di mezzi da una Microsoft che aveva finalmente adottato alla grande l'arma pubblicitaria. Fino al lancio di Windows 95, questa era stata appannaggio della sola creatività Apple. Ma mentre Novell provava ad erodere installazioni offrendo servizi innovativi come il directory service, Linux estendeva e ampliava il mercato potenziale.

Era infatti sufficiente una macchina 386sx, con costi frazionari rispetto alle proposte dei concorrenti, per mettere su un server. Equivalente a NT, anche se all'epoca non erano molti quelli disposti a crederci, ovviamente. La logica prevalente dovuta ad ormai due decenni di incessante legge di Moore (secondo cui la potenza degli elaboratori raddoppia ogni diciotto mesi) avevano abituato il pubblico a sistemi operativi sempre più massicci, con richieste di elaborazione sempre più esose.

Ma un software stabile e scritto bene non ha bisogno di hardware sempre più potente per tenere continuamente in piedi uno

stesso servizio, come invece faceva credere Microsoft. Per condividere un po' di spazio-disco in rete, un piccolo processore è più che sufficiente.

Linux ormai competeva in stabilità con NetWare ed era di gran lunga più avanti di NT. Ransom Love e Bryan Sparks, due tecnici particolarmente svegli di Novell, potevano constatarlo quotidianamente. Scrivere software su Linux era fantastico, disponeva di tutte le astrazioni Unix e addirittura delle stesse interfacce di programmazione (Application Programming Interface, API). Il mondo era inondato di piccoli pezzi di software che risolvevano pressoché ogni problema, bastava saper scegliere ed integrare. Esattamente quello che avevano fatto alla Novell con NetWare e tutti gli altri sistemi operativi. È vero che a volte questo software non offriva un livello di qualità da potenza industriale, e c'era ancora da lavorarci sopra, ma esisteva comunque una buona traccia e non bisognava quasi mai partire da zero. Sapendoci fare e avendo a disposizione venticinque anni di storia della programmazione, come vantavano i laboratori Novell, sarebbe stato un gioco da ragazzi salire in groppa a Linux per la "Total World Domination", come aveva iniziato a proclamare scherzosamente Linus Torvalds.

Il gruppo di progetto spontaneo nato nei laboratori Novell si faceva chiamare, non a caso, Corsair Project, e comprendeva Ransom Love e Brian Sparks sotto la gestione di Rob Hicks, che pur essendo dipendente dell'ufficio legale si trovava sempre in mezzo alle situazioni importanti. L'idea era quella di integrare tutte le tecnologie in mano a Novell per creare un sistema grafico in grado di eseguire contemporaneamente applicazioni Windows, DOS, Netware e UNIX. Un progetto ambizioso, ma Novell aveva in mano tutto quanto serviva per por-

tarlo al successo. L'architettura di massima venne completata con varie realizzazioni tecniche, da un bel nome commerciale, Exposé, e persino da un apposito comunicato stampa:

Novell distilla un nuovo ambiente GUI a 32-Bit – Da *PC Week*, 25 Aprile 1994

Novell sta sviluppando un economico ambiente operativo a 32 bit basato sulla versione “freeware” di Unix che dovrebbe girare – dicono alcune fonti – su Windows, DOS, NetWare e applicazioni Unix.

Novell darà una dimostrazione del software – il cui sviluppo è sotto la stretta sorveglianza di un laboratorio fuori sede – a pochi e selezionati utenti la prossima settimana al *NetWorld & Interop*, affermano fonti vicine a Provo. Secondo queste ultime, il nuovo sistema, nome in codice Exposé, non è un derivato di UnixWare di Novell, ma si basa su Linux, un clone completo di Unix per PC distribuito sotto la libera GNU Public License. Linux 1.0, uscito in marzo, gira su computer ISA e EISA basati su 386 e 486.

Exposé deriva da una versione dell'ambiente grafico X Windows System chiamata Looking Glass, che Novell ha licenziato da Visix Software di Reston. Probabilmente userà un ambiente di lavoro desktop avanzato in 3D per consentirne agli utenti la facile navigazione. Sempre secondo fonti familiari con il progetto, “non è tanto un ambiente applicativo quanto un front end per molteplici ambienti, inclusi NetWare, Unix e Windows”. Gli utenti saranno in grado anche di far girare Exposé come interfaccia per Internet, forse con la GUI di Mosaic.

Tuttavia lo sviluppo sarebbe a uno stadio iniziale e secondo i piani di Novell il progetto potrebbe essere abbandonato qualora non si dimostrasse una solida promessa. Secondo un'altra fonte, Novell avrebbe già dimostrato che la suite Office di Microsoft Windows funziona con Exposé senza dover ricorrere a un emulatore, anche se Linux non supporta pienamente le applicazioni Microsoft.

Lo scopo di Novell, sostengono tali fonti, è di entrare con rapidità sul mercato degli ambienti operativi grafici per offrire agli utenti di PC alternative a basso costo a Windows. L'ambiente dovrebbe costare verosimilmente meno dei 249 dollari di UnixWare e stare sotto anche ai 149 dollari del prezzo al dettaglio di Windows. "Ray [Noorda] lo regalerebbe, se potesse", aggiungono i ben informati.

La licenza GNU consente agli sviluppatori di usare e modificare il codice di Linux e di venderlo a qualunque prezzo di mercato. Con un'avvertenza: insieme ai prodotti derivati, va distribuito anche il codice sorgente di Linux.

Alcuni utenti aziendali di NetWare mettono in discussione la sagacia di Novell nello sviluppare un altro sistema operativo grafico a 32 bit. "Detesto veder spendere un mare di risorse in ricerca per l'ennesimo sistema operativo", sostiene Jim Queen, direttore del settore networking di Enron, azienda energetica di Houston con una vasta rete NetWare. "Se qualcuno ha una specifica visione in materia, meglio farebbe a condividerla". Un altro dirigente informatico spiega di essere alle pre-

se con una serie di sistemi operativi desktop aziendali capaci di cooperare sulla LAN interna. Ma, pur non volendo fare affari con un diretto rivale, “manterrò la mente aperta”, dichiara Lee Roth, manager della Southwest Airlines di Dallas. “Se [Exposé] mi offrirà nuove funzionalità, lo prenderò in considerazione”.

Lo staff del laboratorio discusse a lungo la svolta di Novell verso Exposé basata su Linux, ma non c’era modo di convincere neppure i dirigenti della divisione sviluppo a sostenere una simile posizione. Exposé era pronto per la spazzatura.

Il nuovo management di Novell si era invischiato in Java. Frankenberg, attratto nell’orbita Sun e in particolar modo dalla promessa di Java come linguaggio universale per Internet, aveva lanciato tutte le proprie risorse nello sviluppo di strumenti web. Il progetto Exposé come nuovo sistema operativo diventava di troppo. Al più poteva assumere le sembianze di un browser http per NetWare, da agganciare al progetto di porting su NetWare di Netscape.

Fu Rob Hicks a imboccare proprio questa strada, una volta abbandonato il Corsair Project. Dopo aver contrattato con Netscape i diritti per il porting su NetWare dei programmi di punta (quali Netscape FastTrack Server, Netscape Enterprise Server e Netscape Messaging Server), diede vita a Novonyx, grazie al supporto tecnico ed economico della stessa Novell. La mossa era finanziariamente azzeccata, ma completamente folle dal punto di vista tecnico. Sebbene il server web FastTrack fosse tecnologicamente avanzato e avesse un discreto mercato, oltre che un buon nome dovuto alla fortuna finanziaria di Netscape, il server libero Apache avanzava molto velocemente e,

vista la qualità del prodotto, erano veramente in pochi a dubitare del suo successo. D'altro canto, nel settore dei server email, di cui il Netscape Messaging Server faceva parte, Sendmail era oramai lo standard de facto nel mondo Unix e, ancor più importante, Novell possedeva già la propria suite di programmi di posta. Un pacchetto dalle prestazioni non eccezionali ma neppure scadenti, e che un gruppo di lavoro interno stava integrando in un più complesso sistema di groupware chiamato Groupwise. L'unico prodotto che avrebbe avuto senso era l'Enterprise Server, un sistema web più avanzato, con il supporto di Java e dell'automazione del server.

Novell aveva intanto sviluppato un proprio NetWare Web, con capacità intermedie tra FastTrack e Enterprise, rilasciato gratuitamente con il sistema operativo. Per sostenere la Novonyx, arrivò perfino ad eliminare il proprio NetWare Server dall'offerta gratuita per far spazio al FastTrack Server di Hicks, sebbene questo andasse pagato a parte, alienandosi così una parte dei propri utenti che passarono a Unix o a Linux con il server web libero Apache o Windows NT con IIS gratuito. I clienti fedeli a NetWare si videro costretti a passare a Enterprise Server di Netscape/Novonyx, con costi elevati per via soprattutto delle royalties di Netscape.

Solo nel 2001 Novell riuscì a introdurre nelle piattaforme supportate dai suoi Portal Services una versione per NetWare di Apache. A Hicks fu allora chiesto di dissolvere la Novonyx, venne riassunto in Novell e si portò dietro i diritti dei server Netscape per NetWare. Novell poté offrire gratuitamente sia FastTrack sia Enterprise, che solo qualche giorno prima faceva pagare salato, e fu obbligata a offrire degli sconti sostanziosi, come riparazione ai clienti che l'avevano già acquistato.

Ransom Love e Bryan Sparks capirono subito di non avere molte scelte all'interno di Novell. La strada preparata per Exposé dal management era simile a quella di Novonyx per la realizzazione di un buon browser per NetWare. Verosimilmente si trattava di un progetto ancora meno logico, poiché in pochi avrebbero usato un altro browser incompatibile con Internet Explorer o Netscape.

L'esperienza nell'ufficio legale di Rob Hicks e nella creazione di Novonyx, ricca di luci e ombre, spinse il gruppo a cercare altre strade senza attendersi alcun aiuto da Novell.

Secondo Hicks, già amministratore fiduciario del Noorda Family Trust, c'era però qualcuno che avrebbe potuto essere interessato ad ascoltarli, ma soprattutto a finanziarli: Ray Noorda in persona. Costui non manovrava più le leve di comando dentro Novell, ma era ancora abbastanza influente da poter mettere un'impresa nella giusta luce per allettare i manager che lavoravano per lui. Aveva cresciuto un'intera generazione di rampolli e ne conosceva il metro di valutazione delle iniziative, non foss'altro perché era stato lui a definirne il valore all'interno di Novell. Disponeva di un innato fiuto per gli affari e di parecchio denaro. Se Noorda avesse dato il suo avallo, Ransom e Bryan sarebbe stati pronti a partire in quarta.

Tuttavia il vecchio Ray aveva altre idee per la testa al momento. Non gli interessava competere sul mercato e meno ancora si curava di prodotti innovativi. L'ultimo anno in Novell aveva sancito la sconfitta della sua strategia di scontro muscolare con Microsoft, ma non tutto era perduto. In azienda aveva collezionato alcune perle con l'acquisto di Unix, DR-DOS, WordPerfect e Quattro Pro. Mollato il controllo di Novell, non era detto che non potesse però controllare questi prodotti.

Le idee di Love e Sparks apparivano sufficientemente vaghe e inconcludenti, e la coppia si dimostrava avventata al punto giusto. Noorda, che aveva appena ridipinto la facciata del Noorda Family Trust dedicando il nome Canopy Group alle attività tecnologiche, decise di fornire ai due il capitale d'impresa necessario. La finanza mormone era la prima ad investire su Linux. Love e Sparks erano incentivati ad occuparsi di sistemi desktop basati su Linux attraverso un accordo con RedHat, e di sistemi embedded tramite gli stretti contatti stretti che Noorda aveva con Motorola.

Per quest'ultimo compito, Noorda si era impegnato a convincere Novell a cedergli il DR-DOS, ora ribattezzato Novell DOS, che era ormai un residuo dal valore storico per il mercato dei PC, ma che rivelava un valore non disprezzabile per i sistemi embedded.

Il DR-DOS nasceva nella DRI di Kildall come rimaneggiamento dell'antico CP/M per una sorta di rivincita di Kildall nei confronti di Gates. Quest'ultimo ha sempre accreditato la storia secondo cui non voleva creare quel sistema operativo per IBM, che lo aveva reso ricco e famoso, quanto piuttosto limitarsi alla realizzazione dei linguaggi – motivo per cui aveva passato proprio a Kildall il contatto con IBM. Ma costui non si presentò all'appuntamento e il dirigente IBM ripiegò sulla Micro-Soft (che allora usava ancora il nome originario con il trattino) per il sistema operativo. Non avendo né voglia né capacità per realizzarlo, Gates e Allen, i due soci fondatori, decisero di acquistarne uno già pronto: l'86-DOS di Tim Paterson, originariamente denominato QDOS (Quick and Dirty Operating System).

Kildall ha sempre sostenuto che quest'ultimo non solo fosse una brutta copia del CP/M, ma che contenesse materiale pro-

prietario non licenziato, rubato dal proprio lavoro. E quella non fu certo l'ultima volta che Microsoft si tirò addosso simili accuse. E oltre ad essere peggiore, più lento e meno modulare – insomma poco più di un programma giocattolo – l'86-DOS aveva come unica differenza rispetto a CP/M l'incompatibilità delle applicazioni.

Eppure fu il DOS di Microsoft, prontamente ribattezzato MS-DOS, a diventare lo standard, e alla fine anche Kildall dovette piegarsi a riscrivere il proprio CP/M per essere compatibile con il formato eseguibile scelto da Microsoft. Ne venne fuori il DR-DOS, che però mantenne tutte le peculiarità di modularità e pulizia architetturale che già contraddistinguevano il CP/M, oltre ad una maggiore disponibilità di comandi di utilità e di supporto. E persino una semplice interfaccia grafica chiamata GEM. Tali caratteristiche lo rendevano una piattaforma ideale per l'integrazione nei progetti hardware embedded di Sparks in Caldera, e anche se questa si poneva formalmente come un'azienda pro-Linux, in effetti una parte importante dei propri progetti di sviluppo erano invece basati sul DOS.

Arriva Caldera International

Caldera International Incorporated nacque quindi con l'obiettivo di dimostrare a Novell la fattibilità di Linux come desktop aziendale e sistema di rete. Il prezzo che Noorda chiedeva a Sparks, il quale prenderà inizialmente le redini dell'azienda, era di portare avanti la causa contro Microsoft sull'affossamento di DR-DOS. Si trattava sostanzialmente di una piccola bega locale celebrata in un tribunale dello Utah, in una situazione chiaramente sfavorevole al gigante di Seattle, che proverà a non farsi neppure vedere in aula. Già in Novell, Ray Noorda aveva ottenuto una pronuncia contraria alla pratica commerciale di

Microsoft presso la FTC (la Commissione Federale per il Commercio statunitense), e la condanna di Microsoft sembrava scontata. Ma Frankenberg aveva tutta l'intenzione di offrire un ulteriore regalo all'azienda di Redmond, lasciando cadere la causa in corso. Noorda era ovviamente contrario e decise di proseguire in autonomia, ma con la copertura di Caldera.

Il litigio, trascinato per anni, finì per avere abbastanza poco senso nel contesto generale del mercato e anche la posizione di Microsoft ne sarebbe risultata poco scalfita, se contemporaneamente l'ira di Noorda verso Gates non fosse stata in grado di trascinarlo già davanti l'Antitrust. La bega avrà termine solo qualche anno più tardi con una transazione extra giudiziale di modesta entità e tanta gloria per Caldera nel mondo Linux, mentre il contraccolpo risentito da Microsoft si limiterà a pochi centesimi di dollaro sulle quote azionarie.

In Utah, godendo del clima pesantemente critico nei confronti del gigante di Seattle, i legali e il marketing di Noorda condussero però una vera e propria guerra mediatica, tanto da ventilare richieste fino a 1,5 miliardi di dollari.

Caldera Incorporated vantava sì un buon punto nei confronti di Microsoft, ma nulla di eclatante. Quando quest'ultima decise di metter fine allo stillicidio di informazioni e chiudere il caso con il pagamento di una somma per Caldera, l'entità della cifra non venne resa ufficialmente nota, ma secondo alcune fonti non superava i 60 milioni di dollari.

Noorda assaporò così la sua prima, piccola vendetta. Il colpo grosso lo avrebbe invece sferrato con il governo federale.

Grazie a 1.000 dollari devoluti ufficialmente durante la campagna elettorale e ad altre pressioni, Noorda riuscì ad ottenere dal senatore repubblicano Orrin Hatch una pronuncia storica con-

tro Microsoft. Sebbene Ballmer fosse nel frattempo diventato uno scalmanato sostenitore di Al Gore e Bill Clinton, contribuendo pesantemente alla campagna elettorale dei democratici (e persino spingendo pubblicamente altri ad imitarlo), Microsoft non godeva di grande appoggio con il nuovo governo.

La famiglia Gates era una delle colonne repubblicane dello stato di Washington e difficilmente avrebbero potuto trovare consensi alla Casa Bianca. Tuttavia, il giovane Bill, dichiarandosi apartitico e solo vagamente sotto l'influsso liberal, non suscitava certo grande simpatia nel partito repubblicano.

L'operazione di Noorda, all'ultimo anno di Novell, tesa a dimostrare la scorrettezza commerciale di Microsoft e culminata con la condanna della FTC, ebbe certamente il risultato di rendere evidenti le colpe di Gates, senza però riuscire a superare i dubbi sulla condanna esemplare che lo stesso Noorda pretendeva dal governo.

L'idea comune era che Gates andasse punito per il comportamento anti-competitivo, ma senza rischiare di mettere in crisi il primato degli Stati Uniti nel settore high-tech. E Microsoft era il massimo rappresentante di un tale primato. Non fu quindi strano che l'amministrazione democratica di Clinton e Gore non ponesse ostacoli alla causa Antitrust, né stupì che l'atteggiamento generale fosse insolitamente sotto tono. Quello che più sorprese invece fu la pronuncia anti-Gates di una parte dei repubblicani. Guidati dal senatore Hatch e organizzati dalla lobby di Noorda, un nutrito gruppo di deputati conservatori dell'ovest prese chiaramente posizione contro Microsoft, spingendo ancor più la causa Antitrust avviata dalla controparte politica. Quest'atteggiamento sortì l'effetto di raffreddare quella parte dello schieramento democratico che voleva sostenere

una battaglia partigiana sull'argomento, onde polemizzare con la politica economica dei governi precedenti di Bush e Reagan. La causa Antitrust si trascinò così tra alterne fortune, Microsoft ammise parecchie responsabilità, ma alla fine prevalse il commento di Jerry Pournelle sull'incapacità innovativa dei rivali commerciali nel definire il "comune sentire" dell'americano medio, cosa che lasciò Ray Noorda ancora più furente.

Seguono Caldera Systems e poi Lineo

La visione aziendale di Caldera pareva quanto mai vaga. L'idea di Exposé traballava perché – anche avendo ottenuto qualche successo nell'unificazione dei profili di esecuzione delle applicazioni Unix grazie ai sistemi binari ABI e nella realizzazione di nuovi servizi Linux che avrebbero potuto sostituire quelli di NetWare giungendo al punto di integrare il DOS – rimaneva pur sempre fuori Windows.

Onde utilizzare programmi eseguibili progettati per Windows era necessario clonare le Application Programming Interface (API), ovvero le interfacce che il sistema operativo fornisce alle applicazioni. Questo progetto si rivelò subito assai più complesso di quanto non apparisse inizialmente; laddove Windows 3.1 era una semplice estensione del DOS, le nuove versioni dei sistemi Microsoft erano estremamente complesse, con API scritte appositamente in maniera da complicare la clonazione. Il mercato invece si era stabilmente orientato verso questo sistema operativo di Microsoft, sbaragliando praticamente ogni concorrenza. Proporre un sistema in grado di eseguire il software di nove sistemi diversi su dieci, quando proprio l'ultimo era quello che possedeva i nove decimi delle applicazioni, non attirava affatto l'attenzione di eventuali investitori.

Così Caldera si trovò tra le mani un prodotto poco attraente e sulle spalle una responsabilità tremenda: riuscire dove avevano fallito tutti gli altri, inclusa IBM con il sistema OS/2.

Neppure la comunità degli investitori mormoni dello Utah, ampiamente disposta a sostenere le iniziative locali, credeva seriamente in un simile progetto e stroncò senza appello il gingillo di Noorda, che fino a quel momento aveva in realtà prodotto soltanto una rancorosa causa contro Microsoft.

A quel tempo, infatti, le velleità dei produttori di sistemi operativi erano ormai tramontate per tutti: Commodore, Atari, Acorn, che pure avevano tentato di lanciare sistemi certamente avanzati e probabilmente migliori di Windows, coprivano percentuali di mercato decisamente minime.

Senz'altro migliore lo stato di salute di Apple, ma la comoda nicchia ricavatasi in ambito grafico e didattico non sarebbe riuscita a farle conquistare spazi superiori al dieci per cento.

Qualcun altro, pur partito con idee grandiose e risorse finanziarie quasi infinite, come Steve Jobs estromesso dalla stessa Apple, con la NeXT non era riuscito ad affacciarsi seriamente sul mercato neppure grazie al sostegno di Ross Perot in persona.

Le velleità di Caldera apparivano sinceramente esagerate, e non di poco. Ma ancora peggiore era il rapporto tra Caldera e la comunità del software libero e open source. Love e Sparks venivano etichettati come i “nuovi arrivati”, privi di una vera cultura della condivisione. Caldera, che pure contribuiva ad alcuni progetti pubblici, non sempre giocava pulito. Molti criticavano la scelta di John Terpstra, in precedenza vicepresidente di TurboLinux e fondatore del progetto SAMBA, come responsabile dell'ufficio tecnologico.

Sotto la guida di Sparks, l'azienda sfruttava invece al meglio i

contatti di Noorda nel settore dei sistemi personalizzati. L'uso del DOS sembrava essere vincente e, anche se Caldera andava attirandosi le critiche del mondo Linux, l'attività andava talmente bene da poterne capitalizzare i risultati. Nel 1998 verrà creata un'apposita divisione interna denominata Caldera Thin Clients. Nel complesso, tuttavia, l'azienda andava a rotoli e poco prima di finire al macero se ne sganciò la costola di Sparks, cambiando nome in Lineo Inc.

Eccoci così di fronte alla prima morte – e alla prima, immediata rinascita – di Caldera, tramutatasi in Lineo per seguire il destino di Sparks.

Caldera Systems Inc. nacque nel corso di questa prima ristrutturazione di Caldera Inc., ed assorbirà tutto quello che Lineo non aveva portato via con sé, lasciando la società originaria un guscio vuoto, senza né dipendenti né beni, pronto per la liquidazione – ma non prima di aver riscosso l'importante pagamento della causa Microsoft nel 2000.

Caldera Systems venne allora affidata interamente a Ransom Love. Missione aziendale: la realizzazione di desktop Linux e l'esemplificazione di servizi professionali per piccole e medie aziende basati su Linux. Va detto subito che, però, non riceverà molte soddisfazioni dal mercato dei pacchetti Linux. La nuova struttura era di dimensioni alquanto ridotte, come già accaduto con Caldera Inc, ma non occorre certo molte persone per gestire l'assenza di un prodotto. Analogamente, limitato era il contributo complessivo dell'azienda. Si sostanzialmente quasi completamente nella realizzazione di un unico programma: un'inezia, al confronto della quantità di codice distribuito da Caldera Inc. e sul quale non aveva alcuna proprietà.

Quest'unico programma aggiunto dalla nuova Caldera rivestiva

però molta importanza: Lizard era un – anzi, il primo – vero installer di sistema operativo. Per chi usa il PC, l'installazione del sistema operativo è sempre stata un'operazione banale. Fino all'arrivo di Windows 3.11 l'intero sistema poteva essere distribuito su un paio di dischetti che contenevano i programmi principali, tra cui l'interfaccia a linea di comando, l'insieme dei driver per i dispositivi hardware e poco altro. Windows vi aggiunse un'interfaccia grafica, laddove esistevano già alcune alternative semi-grafiche come la shell avanzata 4DOS o grafiche come il sistema GEM, ultima fatica di Kildall, da integrare in DR-DOS.

L'arrivo dei primi hard disk aveva semplicemente significato l'uso di un diverso dispositivo di registrazione, ma nulla era cambiato nel sistema operativo che li gestiva.

Lo stesso NetWare non si discostava granché da questa strategia di funzionamento, limitandosi a nascondere dietro l'uso di due programmi specifici: un sistema di installazione guidata e un sistema di configurazione e manutenzione. Nessuno di questi due programmi risultava particolarmente rifinito o attraente dal punto di vista grafico. Eppure si dimostravano efficaci. Unix era invece basato su una concezione completamente differente. Da sempre pensato per risiedere stabilmente sui dischi fissi dei grandi sistemi dipartimentali, aveva una procedura d'installazione ampiamente adattabile a classi di hardware molto vaste. Il costo di questa flessibilità stava in una notevole complessità nella procedura d'installazione, realizzata tramite un lungo insieme di operazioni da realizzare in sequenza.

Per fare un solo esempio, mentre tutti i sistemi operativi discendenti dal DOS potevano utilizzare il disco rigido in un unico formato, e ancora oggi non sono più di un paio, Unix supporta alcune dozzine di differenti formati per registrare l'informa-

zione ricorrendo a schemi di suddivisione pressoché infiniti. Lizard era un programma semplice, se paragonato ai moderni sistemi di installazione Linux presenti nelle distribuzioni più note quali Mandrake o SuSE. Il vantaggio principale era quello di far passare la macchina in modalità grafica prima ancora dell'inizio dell'installazione, lanciando una versione di Linux ridotta direttamente dalla RAM, limitandosi a porre le identiche domande di altri installer basati su sistemi testuali, attraverso un'interfaccia colorata con scelte multiple gestibili dal mouse.

Nonostante questo sforzo, Caldera Systems Inc. appariva una società tutt'altro che appetibile. In mancanza di un prodotto vero e proprio, e puntando a un mercato ancora in gran parte inesistente, si può ragionevolmente affermare che i beni e le proprietà tangibili dell'azienda fossero praticamente nulli. Convincere qualcuno ad investirvi sarebbe risultata un'impresa ardua persino per vecchie volpi della finanza come Ray Noorda o Rob Hicks. Soltanto qualche folle l'avrebbe fatto.

Ma ad un bel momento... ecco la follia generale. Caldera entrò nel turbine della new economy in una situazione tanto disastrosa che, pur essendosi pubblicamente dichiarata in vendita, tutti sembravano snobbarla.

Paradossalmente era proprio la storia di Caldera a rappresentare un problema per gli investitori, gli stessi che altrimenti non avrebbero esitato a coprire di denaro un'azienda appena concepita e senza nessuna concreta dimostrazione di poter reggere l'urto della competizione.

Nei cinque anni precedenti, Caldera aveva già dimostrato di non essere in grado di raggiungere gli obiettivi preposti. Aveva una bella confezione per un prodotto senza mercato, abba-

stanza rabberciato e privo di alcun diritto proprietario. Troppo sfacciato persino nel clima della New Economy.

Quando alla fine si riuscì a offrirla sul mercato, l'IPO (Initial Public Offering, l'offerta pubblica di azioni alla borsa statunitense) di Caldera ebbe una performance veramente magra: il prezzo del titolo raddoppiò. Da notare che le azioni delle IPO RedHat o VA Linux nello stesso periodo avevano guadagnato il 700 per cento in 6 mesi.

Una fortuna comunque insperata per Love e Noorda. L'azienda, rimaneggiata per l'occasione, superò l'IPO in modo traballante, senza mancare di tirarsi addosso una serie di rogne legali. I soldi che andavano riversandosi su nomi tipo RedHat, nello stesso periodo, erano un fiume in piena e solo qualche schizzo finì per lambire Caldera. Ma era pur sempre più di quanto si potesse immaginare.

Il gioco delle tre carte, iniziato con la costruzione di Caldera Inc, poi diventata Lineo, aveva fruttato al Canopy Group di Noorda la cospicua somma sborsata da Motorola per il successivo acquisto di Lineo, la quale aveva bruciato - allo scopo di farsi acquistare da qualcun altro - quasi 100 milioni di dollari degli investitori, ma soprattutto i milioni che Microsoft aveva sborsato per chiudere la causa di scorrettezza commerciale rispetto al DR-DOS.

Noorda era pronto a ripartire con una Caldera System Inc. nuova di zecca, in cui acquisire gli investimenti di Novell, Sun Microsystems e Citrix, indotti a pompare denaro sull'onda emotiva delle grandiose IPO di RedHat e VA Linux.

Dopo l'offerta pubblica di azioni, Caldera si ritrovò piena di crediti, ma senza nessun vero prodotto. E quel che è peggio, priva di prospettive commerciali. Le idee orientative di Love e

Sparks su Exposé erano definitivamente tramontate e il mercato non richiedeva certamente una distribuzione Linux che per anni era stata il fanalino di coda delle già riscate vendite sugli scaffali dei rivenditori.

Caldera, che pure nel tempo aveva contribuito alla realizzazione di importante software open source grazie a un piccolo laboratorio di sviluppo, non vantava tuttavia né un prodotto né un mercato, non aveva una rete di vendita né una struttura di supporto per dei servizi professionali che peraltro non offriva. A quel punto aveva solo tanti soldi (virtuali) in cassa.

Parte la saga di Santa Cruz Operations

Quando nel 2000 Doug Michaels e Caldera Systems avviarono i primi rapporti, le due strutture si capirono al volo. La Santa Cruz Operations Inc. era una sorta di 'miracolata' di Microsoft, un microbo aziendale la cui fortuna era stata quella di associarsi con il colosso di Redmond quando questo aveva iniziato a offrire un sistema operativo per il mercato professionale.

Nel 1979, infatti, ancor prima della nascita di MS-DOS, Microsoft si lanciò nel mercato dei sistemi operativi professionali acquisendo da AT&T i diritti di Unix con l'obiettivo di realizzarne una propria versione.

AT&T non le vendette però il diritto a usare il nome Unix e, per non incorrere in guai legali, Microsoft decise di denominare il proprio prodotto XENIX. Ciò nonostante, nessuno all'interno di Microsoft aveva mai visto una sola riga del codice Unix. Nel 1979 tutto quello che esisteva di XENIX in Microsoft non era altro che una *brochure* argentata con cui Bill Gates andava in giro per convincere la clientela di essere all'altezza nella fornitura di "sistemi operativi seri".

Il codice AT&T fu in realtà trattato dalla Human Computing Resources (HCR) di Toronto, che (si dice) è stata la prima azienda a realizzare un vero e proprio porting completo commerciale di Unix, avendo rilevato in precedenza i diritti da AT&T come entità autonoma dell'Università di Toronto, da parte del professor Baecker, uno dei primissimi collaboratori nel progetto di Thompson e Richie.

La distribuzione HCR era disponibile sui processori originali PDF/LSI-11 e su quelli 68000. Lo XENIX non sbarcò in USA tramite Microsoft, ma prima con MSD e successivamente nella catena di negozi per elettronica Tandy/Radio Shack. Era installato su dischi da 20Mb con piatti da 14 pollici e girava in 256Kbyte di memoria, una dotazione hardware spropositata per l'epoca.

Fu solo il 25 agosto 1980, in anticipo di quindici anni sui tempi della propria azienda e in ritardo di circa due sulle prime pubblicità, che Bill Gates annusò la possibilità di vendere un sistema operativo serio ai propri clienti, decidendo il rilascio di XENIX.

Microsoft non fu l'unico rivenditore di XENIX negli Stati Uniti, né il primo. E neppure s'impose come l'azienda che riuscì a piazzarne la maggiore quantità di copie, perché la palma d'oro venne conquistata alla grande da Tandy/Radio Shack con il TRS-16 basato su processori 68000.

Intanto, sempre nell'anno di grazia 1979, alcuni dipendenti della TRW-Vidar, piccola società che navigava nel florido mercato delle apparecchiature telefoniche di commutazione di Mountain View, in California, lasciarono l'azienda sotto la guida di Larry Michaels e del figlio Doug per trasferirsi poco più a sud, in quel di Santa Cruz. Qui si dedicarono alla realizza-

zione di sistemi collaterali al mercato telefonico, fondando la Santa Cruz Operations Inc., che ben presto si lanciò nella realizzazione di un sistema Unix esclusivamente dedicato ai processori Intel.

L'11 agosto 1999, trascorsi vent'anni, Larry Michels morì per un secondo ictus. Per la Santa Cruz Operations Inc. fu una perdita molto grave: l'azienda era guidata in modo paterno e senza grandi concessioni alla rapacità capitalistica imperante. Come una famiglia – con tutti i pro e i contro di tale gestione.

Riprendendo quanto ha sostenuto Steven D. Beedle, responsabile di *SCO World*, la rivista sul mondo SCO in edicola nell'agosto 1999, in un fondo dal titolo "Caro Larry Michels": "Hai vestito SCO come uno sciatore che vince la gara scendendo come un indemoniato senza controllo. Non era facile lavorare per te e ancora più difficile era accontentarti. Come leader dell'azienda, eri instancabile nell'ambizione commerciale e nella guida alla conquista della gente. Hai prosperato sottraendo terreno a un concorrente di SCO dopo l'altro, da Microsoft a Microport. Hai portato SCO a sedersi al tavolo con entità del calibro di AT&T, Digital, IBM, Sun e altri grandi. E ti sei assicurato che nessuno potesse dimenticare questa esperienza". Aggiunge Doug Michels: "È stato un lungo e profondo percorso iniziato con lo shock e il terrore del primo colpo, attraverso il miracoloso processo di recupero e l'inaspettata conclusione". Un servizio commemorativo avrà luogo per celebrare la vita di Larry, a Santa Cruz, California, il 20 novembre."¹

Sebbene la Santa Cruz Operations Inc. potesse dimostrarsi un posto piacevole dove lavorare (e la caffetteria pare non avesse rivali in tutta la zona), non si trattava certamente di un'a-

¹ Tratto da: <http://www.maxframe.com/NEWS.HTM>

zienda particolarmente avanzata né ben valutata nel proprio settore.

Nel 1987 SCO aveva scambiato con Microsoft i diritti di distribuzione di XENIX per i sistemi Intel, cedendo una quota consistente dell'azienda e con un rappresentante Microsoft come presenza stabile nel consiglio di amministrazione. Successivamente Microsoft vi investì una somma cospicua acquistando quasi il 12 per cento delle quote societarie per controllarne l'operato all'interno di X/Open, il consorzio dei produttori Unix.

L'ingenuità manageriale della famiglia Michels era facilmente orientata dal gigante di Seattle, che negli anni Ottanta vedeva in SCO un comodo alleato su cui scaricare quei problemi gestionali di XENIX come sistema professionale di fascia intermedia che Microsoft, non producendolo, non riusciva a gestire.

La dimensione aziendale di SCO e l'incapacità degli amministratori erano tali da non incutere timore, e Microsoft avrebbe potuto fagocitarla in qualsiasi momento. Nel frattempo Unix era stato abbandonato dalla stessa Microsoft non per le qualità tecniche, che anzi erano comunemente apprezzate anche da Bill Gates,² quanto piuttosto per i problemi relativi ai diritti commerciali che non avrebbero mai potuto garantirle il monopolio nel campo. Infatti AT&T dava l'idea di non voler cedere a nessun costo i diritti complessivi del codice (come poi farà solo dieci anni più tardi con Novell).

SCO rimase così a lungo sotto l'ombrello protettivo di Microsoft, che paradossalmente resterà invischiata nell'utilizzo di XENIX nella propria rete aziendale per un lunghissimo perio-

² Si veda: <http://microsoft.com/billgates/speeches/industry&tech/uexpo.asp>

do anche dopo l'uscita del proprio sistema server di bandiera, Windows NT.

La prima visione aziendale di Larry e Doug Michels sarebbe stata esplosiva, ma si scontrò con due grossi problemi.

Il primo era un problema prettamente tecnologico: XENIX prima e SCO Unix poi, diventato più tardi OpenServer, erano ben al di sotto dei livelli di qualità industriale dei sistemi Unix, secondo quanto ritenevano gli stessi rivenditori. Il supporto tecnico periferico dell'azienda era di livello appena decente, mentre quello centrale dei laboratori SCO al massimo poteva dirsi sufficiente. Installatori e rivenditori a valore aggiunto dovevano spesso inventarsi soluzioni senza attendersi quasi nulla dall'azienda che, soprattutto in Europa, agiva da semplice rivenditore di pacchetti software.

Il secondo problema era Microsoft stessa, che aveva sicuramente un effetto benefico sull'azienda, assai superiore alla mera presenza nel consiglio d'amministrazione. Non a caso SCO veniva identificata un po' da tutti con XENIX, e questo a sua volta con Microsoft. La forza di quest'ultima nel settore dei PC andava riflettendosi sui sistemi professionali, anche quando non vi compariva direttamente con il marchio aziendale. I produttori di applicazioni, inizialmente per l'induzione causata dalla stessa Microsoft e in seguito per scelta autonoma, avevano proclamato XENIX, e subito dopo SCO Unix, il sistema operativo prescelto da piccole e medie industrie.

Lo stretto rapporto con Microsoft offriva a SCO il vantaggio di un vasto parco di applicazioni, vivendo però sotto un'amministrazione praticamente controllata dal gigante di Redmond. Era chiaro che se e quando Microsoft si fosse decisa a competere con SCO, questa si sarebbe ritrovata seriamente nei guai.

Così avvenne. Con l'avvento di Windows NT, che entrava in competizione diretta con SCO Unix, il comodo mercato di nicchia per Santa Cruz Operations semplicemente svanì. Con l'arrivo in campo di Microsoft, non esisteva più la necessità di portare le applicazioni su Unix. In maniera lenta ma drastica, il mercato di SCO andava assottigliandosi a vista d'occhio. E ciò a prescindere da Linux.

Oggi The SCO Group, la società che ha rilevato le attività Unix di Santa Cruz Operations, tenta di incolpare Linux per la perdita di smalto subita da SCO Unix. Ed è vero che Linux si poneva come chiara alternativa a SCO nei sistemi basati su processori Intel. Tuttavia il declino di SCO sul mercato fu chiaramente dovuto all'introduzione di Windows NT ben prima della penetrazione Linux nel medesimo settore. Altrettanto vero è che oggi nessuno penserebbe di installare un sistema Unix, optando per SCO al posto di Linux. Ma dieci anni fa la situazione era ben diversa, e numerosi clienti SCO passarono a Windows NT, non certo a Linux.

D'altro canto, Microsoft non aveva alcuna intenzione di distruggere SCO. Quest'ultima svolgeva infatti un importante servizio: con l'acquisizione dei diritti sul codice Unix di AT&T, SCO costituiva di fatto una fonte di controllo su produttori come Sun, IBM o Hewlett-Packard (HP). I contratti di tali produttori non erano così stringenti e non richiedevano pagamenti esosi perché, in un modo o nell'altro, quasi tutti si erano affrancati dal giogo della proprietà intellettuale di AT&T, accollandosi dei costi di uscita perenni. Inoltre SCO non vantava alcun controllo sul presente e sul futuro di Unix, poiché il copyright sul marchio Unix, e le complesse procedure secondo cui un sistema potesse qualificarsi come sistema "compatibile con

Unix”, erano gestiti dal consorzio di produttori Unix X/OPEN. Ciò nonostante, nessuno di quei produttori avrebbe mai potuto sentirsi completamente libero di usare Unix a proprio piacimento.

La causa SCO contro IBM va dunque vista tutta all’interno di questo micromondo degli Unix proprietari.

Infine, SCO era l’unico produttore proprietario di Unix per sistemi Intel e poteva quindi pretendere di sedersi, con la stessa autorevolezza, al tavolo della standardizzazione Unix con IBM, Sun e HP. Se non altro per riferire a Microsoft quanto avveniva in quell’ambiente³.

Tuttavia, nel complesso SCO risultava un’azienda mal amministrata, con un prodotto obsoleto e un avvenire quanto mai incerto. Doug Michels prese il posto del padre nell’amministrazione societaria, adottando una modalità di comando “per insulti ed intimidazioni” che indusse gran parte delle risorse migliori, sia tecniche che manageriali, a ritirarsi.

Michels decise allora di tirare i remi di Unix in barca, diversificando la produzione — sia per sottrarsi al giogo di Microsoft, sia perché lui stesso non vedeva un grande futuro per lo SCO Unix. Come potevano pochi programmatori a Santa Cruz competere con il mondo dei programmatori Linux?

Poiché le capacità tecniche certo non mancavano, ci si concentrò nella realizzazione di software intermedi per la connessione dei vecchi sistemi che ancora permanevano nei servizi informativi interni, e che dovevano trovare il modo di essere esportati verso le Intranet aziendali o addirittura su Internet.

Questi software di piccole dimensioni, la cui raccolta veniva denominata Tarantella, risolvevano tutta una serie di problemi di con-

³ Si veda: http://www.findarticles.com/cf_dls/m0CGN/n3395/20511132/p1/article.jhtml

nessione. E pur essendo molto meno blasonati di un intero sistema operativo, si dimostrarono una buona fonte di reddito.

Santa Cruz Operations Inc. entrava così in un mercato completamente differente da quello della vendita dei sistemi operativi, finendo per riorganizzarsi in tre distinte unità operative: la Server Software, che si occupava del supporto tecnico e dell'evoluzione degli Unix; il settore Professional Services, per la gestione dei servizi correlati alla vendita delle licenze, come manutenzione e installazione; e una terza unità centrata sul pacchetto Tarantella, ormai divenuto la principale fonte di entrate per l'intera struttura. Le rendite del sistema operativo erano infatti in caduta libera. Nel 2000 Michels si trovò di fronte anche al cospicuo disinvestimento di Microsoft, che decise di abbandonarlo provocando perdite ingenti nella società, tanto da obbligarlo ad uscire dal mercato azionario statunitense. Lo fece dissolvendo la vecchia azienda e il relativo titolo SCOC per rifondare la Tarantella Inc., in cui la divisione Server Software era più un'entità formale che sostanziale, limitandosi a sporadici rilasci di supporto di sicurezza per i due sistemi operativi di casa. Nel 2001 non esistevano due aziende più diverse tra loro di Caldera Systems Inc. e Santa Cruz Operations Inc., che in realtà aveva già cambiato nome in Tarantella Inc. e non lasciava adito a dubbi sulla volontà di liberarsi delle proprie versioni Unix.

La prima, va ricordato, aveva incamerato un'enormità di denaro fresco dagli investitori senza possedere alcun prodotto o mercato tangibile, mentre la seconda si trovava ad avere tra le mani le vestigia di una struttura di vendita calibrata per distribuire a livello mondiale un sistema operativo di larga diffusione, ma che in realtà operava per piazzare un semplice middleware ben fatto, senza troppe pretese, con pochi, ben selezionati e danarosi clienti.

Il fatto che Santa Cruz Operations Inc. dovesse finire sotto un qualche tipo di acquisizione era così evidente nel 1999 che persino in Italia il primo editoriale della neonata *Linux Magazine* ne ventilava la concreta possibilità:

“...per l’inspiegabile euforia borsistica sui tecnologici che porta RedHat ad avere una capitalizzazione più di 10 volte superiore a SCO in tempi incredibilmente brevi (a quando l’Offerta Pubblica di Acquisto?)”.

E così fu – quasi.

SCO tentò di farsi bella agli occhi Caldera, visto che già anni prima aveva raziato l’open source realizzando una distribuzione per il proprio sistema chiamata Skunkware, molto apprezzata da una clientela abituata a una qualità tecnica di certo inferiore. Ma nel periodo precedente all’acquisizione da parte di SCO, Michels non fece segreto di voler contribuire a Linux con le proprie tecnologie. Ma neppure voleva finire nelle fauci di qualche pescecane, onde continuare a gestire l’azienda con il piglio padronale a cui era abituato. Ransom Love ricorda come fosse proprio una simile gestione a farlo inorridire, e non avrebbe mai accettato che un’unità operativa venisse modellata come una casa-azienda a conduzione familiare. L’accordo finale stabilì così lo scorporo dei prodotti server da Tarantella: più che di una acquisizione societaria, si dovrebbe quindi parlare di una vendita vera e propria.

Si tratta di un dettaglio non trascurabile. Tra l’antica e gloriosa SCO del passato e la nuova SCO non esiste alcuna relazione: sono due società differenti, nei dirigenti che la compongono e nella mentalità che la anima.

L'attacco di SCO e McBride

Proseguendo l'exkursus storico, nel 2001 Santa Cruz Operations non aveva più i requisiti per essere quotata in borsa e Doug Michels fece in modo di trasformare la società in Tarantella Inc. Questa passò dalla quotazione al Nasdaq Stock Market con simbolo SCOC, a quello a più piccolo e meno blasonato del Nasdaq SmallCap Market, l'equivalente del Nuovo Mercato in Italia, con il simbolo TTTL.

Contemporaneamente sul versante Caldera, Ralph Yarro e Ransom Love creavano una nuova struttura finanziaria identificata generalmente come Caldera Holding Inc. e presto rinominata Caldera International Inc. che avrebbe gestito i diritti del codice Unix AT&T (ora UnixWare), e quelli del vecchio SCO OpenServer, successivamente identificato come SCO Unix (e che una volta era semplicemente Xenix). OpenServer però rimaneva di proprietà di Tarantella Inc., a cui Caldera, unico distributore, avrebbe versato gran parte degli eventuali ricavi di vendita. Caldera Holding/International manteneva anche la distribuzione Linux Caldera OpenLinux.

Dei tre prodotti, OpenServer sarà considerato così poco interessante che Caldera tentò persino di mollarlo a Tarantella, svincolandosi dall'obbligo di doverlo supportare. Lo rileverà completamente solo l'anno successivo a seguito del disinvestimento dovuto alla riduzione del debito aziendale. OpenServer è oggi il prodotto che produce i maggiori ricavi per SCO.

Intanto il simbolo SCOC veniva estromesso dal Nasdaq Stock

Market per assenza dei requisiti minimi. The SCO Group Inc. vi sarebbe rientrata solo qualche anno più tardi come SCOX, sfruttando le risorse finanziarie dell'IPO di Caldera.

Fu così che Santa Cruz Operations venne lasciata morire, e anche Caldera passò a miglior vita per la seconda volta in due soli anni, lasciando in eredità al Canopy Group i propri beni patrimoniali, in parte utilizzati per mettere in piedi la nuova holding Caldera. Stranamente tra l'annuncio e la realizzazione pratica dell'affare trascorse un intero anno. Fu infatti nel 2000 che Caldera Inc. (da non confondere con Caldera Systems Inc.) si spostò dallo Utah allo stato del Delaware, cambiando nome in Caldera International, per tornare poi in Utah l'anno successivo identificata come Holding Caldera Inc., ovvero la "Newco" citata nei documenti ufficiali della vendita tra Santa Cruz Operations e Caldera Systems Inc.

La logica di questo immotivato trasferimento va ragionevolmente cercata nel fatto che in Delaware è possibile godere di normative sui passaggi societari particolarmente restrittive, impedendo le cosiddette 'scalate ostili'. Un'azienda va necessariamente acquisita in blocco, non basta lanciare una offerta pubblica di acquisto. Una limitazione che oggi torna quanto mai utile a SCO. Dopo l'operazione rimase, esempio classico dei giochi di Ray Noorda, una Caldera Systems Inc. vuota, senza dipendenti né beni, il cui capitale – dovuto in parte all'IPO – poteva essere liquidato a piacere degli amministratori rimasti. Nella nuova Caldera International entrerà a far parte del consiglio di amministrazione Ralph Yarro, una figura che, pur rimanendo nell'ombra lungo il corso dell'intera vicenda, risulterà centrale. Sarà infatti lui il factotum al servizio di Ray Noorda, fino a divenirne amministratore fiduciario di tutte le attività di inve-

stimento, nonché presidente, consulente o consigliere di una infinità di aziende legate agli interessi di Noorda – molte delle quali diretti investitori nel business SCO. Nel 1994 Ralph Yarro è un oscuro artista grafico che realizza sfondi tridimensionali per un videogioco senza successo. Nel 1995, a 26 anni, diventa l'amministratore fiduciario di una delle più ingenti fortune del capitalismo statunitense: il Noorda Family Trust. Strano, no?

A valle del canale

Dopo la fusione in Newco, i rivenditori SCO, spossati da due anni di completa latitanza della società, tirarono un sospiro di sollievo. Finalmente esisteva un piano di migrazione realistico verso un sistema Unix futuro che li rimetteva in gioco. Molti si rimisero all'opera per tentare di diversificare la clientela lungo le due linee di prodotti principali: Caldera Linux, come sistema professionale per workstation o piccole installazioni server e SCO UnixWare come sistema professionale per installazioni più consistenti a più alta affidabilità o per multiprocessori, dove Linux aveva ancora delle limitazioni. Ai rivenditori veniva chiesto di proporre subito la migrazione degli SCO Unix (quello che proveniva dal vecchio Xenix ed era considerato comunemente di pessima qualità) in Caldera Linux, su cui Caldera di fatto non guadagnava nulla.

Un sollievo per lo più ingiustificato: furono pochissimi i clienti che si fecero convincere a passare al costoso UnixWare, mentre molti colsero l'occasione per liberarsi del vecchio UNIX SCO per scegliere Linux Caldera, e alcuni, soprattutto in Europa, direttamente SuSE.

I primi passi della nuova Caldera International furono quanto mai incerti. Si assisteva ad un profondo scollamento tra le idee

– e gli interventi pubblici – di Ransom Love, che pure da Presidente avrebbe dovuto contare qualcosa nell'azienda, e la volontà di tutti gli altri attori della vicenda.

Love aveva ora tra le mani alcune tecnologie interessanti per Linux. Innanzitutto c'era il sistema di clustering integrato in UnixWare, chiamato Non Stop Clustering, un punto importante nella competizione con RedHat, la quale proponeva un sistema simile, ma più giovane e di gran lunga meno blasonato. Esisteva poi il progetto Monterey, a cui SCO aveva partecipato insieme a IBM, Intel e Sequent. Sarebbe stato un buon colpo per Linux acquisire il supporto per i processori IA-64 proveniente da SCO. Come comunemente accade in questi casi nell'open source, l'azienda avrebbe acquisito un know-how fenomenale che le avrebbe garantito una comoda posizione di rendita.

Il sistema di clustering adottato dallo SCO UnixWare era composto da un insieme molto vasto di programmi di connessione, con il compito di permettere la sostituzione 'a caldo' di una macchina non più disponibile, ad esempio per un guasto, con un'altra precedentemente configurata per agire in sostituzione. Questo sistema permette di aumentare la disponibilità di un gruppo di macchine (un cluster, appunto) svincolandolo da possibili problemi operativi. Il sotto-sistema NSC di UnixWare era tradizionalmente considerato molto affidabile e migliore, ad esempio, dell'equivalente HACMP disponibile sulla piattaforma RS/6000 per Unix AIX di IBM.

Purtroppo NSC non era affatto di proprietà SCO, ma venne sviluppato da quest'ultima per conto di Compaq, basandosi su una precedente realizzazione di Tandem. NSC imponeva l'uso di schede proprietarie Tandem e i diritti di SCO terminavano con la possibilità di includerlo assieme alle installazioni di

UnixWare, non certo di poterne fare quello che voleva a proprio piacimento. Nel 2000, in una mossa di chiara ostilità nei confronti di Compaq che aveva nel frattempo acquisito Tandem, SCO mise a punto un sotto-sistema di connessione che rendeva inutili le schede proprietarie Compaq/Tandem, onde poter integrare NSC in modo generico in UnixWare con semplici schede Ethernet a basso costo. Ma il nucleo della tecnologia era rappresentato dal sotto-sistema SSI (Single System Image) sviluppato da Compaq. In definitiva, senza l'accordo di Compaq, SCO poteva vantare diritti sulla metà del sistema, e neppure quella più importante. Tanto è vero che Caldera, una volta acquisito UnixWare, si decise ad estirpare dal kernel il sotto-sistema di clustering, in modo da non pagarne più le royalty. Compaq poté renderlo così disponibile con una licenza open source, creando addirittura una sezione apposita su SourceForge per i propri progetti NSC e SSI, e ottenendo tutta la facile pubblicità che Caldera avrebbe voluto per sé.

Sulle colline di Monterey

Monterey invece sembrava essere uno di quei progettioni senza speranza nati per seguire l'avanzata tecnologica imposta da Intel per i processori a 64 bit.

L'evoluzione degli Unix proprietari di IBM (AIX) e SCO (UnixWare e OpenServer) doveva passare sicuramente attraverso un'analogia evoluzione della piattaforma hardware di esecuzione. I processori a 32 bit, pur avendo mostrato di essere ampiamente sufficienti nella realizzazione di comuni servizi info-tech, avevano la propria naturale prosecuzione in quelli a 64 bit, i quali presentano quantomeno minori limiti di indirizzamento dell'informazione.

Intel, maggior produttore di processori al mondo, aveva deciso di evolvere la propria piattaforma a 32 bit, il noto Pentium, in una nuova piattaforma a 64 bit, chiamata Itanium e spesso semplicemente IA-64. Per motivi strettamente commerciali e basandosi sulla propria posizione di forza nel mercato, Intel aveva tentato di imporre una completa rivoluzione della piattaforma, la quale non sarebbe risultata compatibile con i Pentium in circolazione. In tal modo avrebbe potuto tenere in piedi due linee commerciali alternative.

L'interesse di SCO nel progetto Monterey appariva evidente: qualcuno avrebbe finanziato l'evoluzione dei propri Unix – una tendenza ad appoggiarsi a finanziamenti esterni che le sarà fatale nella relazione con IBM.

Il sistema AIX di quest'ultima, invece, era saldamente legato alla piattaforma PowerPC, a quel tempo sotto il cappello di Motorola, secondo produttore di chip dietro Intel. IBM tentò quindi di sfruttare l'occasione per avviare il 'porting' verso un nuovo processore, in modo da non rimanere tecnologicamente bloccato con Motorola.

In realtà questo complicato gioco di alleanze costrinse ogni attore a entrare in una modalità di perenne attesa, nella continua aspettativa delle mosse altrui. IBM, nel periodo peggiore della propria crisi aziendale, sperava che SCO si assumesse l'onere di portare a IA-64 un codice solido e stabile come AIX. Da parte sua, SCO non disponeva delle risorse – e forse neppure delle capacità – per affrontare da sola un compito così arduo. L'unica ad affrontare la questione in termini produttivi fu Sequent, che migliorò in modo incredibile le capacità del proprio sistema NUMA.

L'alleanza entrò dunque presto in crisi, ma ciò importava poco ad IBM, il cui vero obiettivo era quello di completare la pro-

pria offerta Unix in modo che fosse scalabile da un comune processore fino agli enormi mainframe.

Fu subito chiaro che gli Unix di SCO non potevano coprire la fascia bassa dell'offerta IBM: tecnologicamente troppo arretrati e realizzati in modo troppo pasticciato per essere considerati nelle linee di prodotto di Big Blue. Inoltre, SCO era dichiaratamente una società inaffidabile e parecchie delle migliori tecnologie prodotte di fatto non le appartenevano completamente – vedi il sistema di clustering NSC licenziato da Tandem/Compaq. SCO aveva raggiunto qualche successo solo grazie ai costi sotto la media e all'ala protettiva di Microsoft. Non ultimo, c'era Linux che avanzava a grandi passi proprio in quegli ambiti in cui già operava SCO.

Non tardarono invece a concretizzarsi gli accordi tra Sequent e IBM, quanto mai produttivi per quest'ultima che, nella propria offerta Unix presentava un'ampia falla. Infatti, tra la piccola piattaforma Netfinity fino a quattro processori Intel (tuttora coperta da Windows ma in prospettiva ben servita da Linux), e i grandi sistemi RS/6000 con AIX si apriva uno spazio per l'architettura NUMA (Non-Uniform Memory Array) di Sequent, in grado di scalare fino a 256 processori. Nel settembre del 1999, con SCO completamente fuori dai giochi di Monterey, si concluse così la fusione tra IBM e Sequent. Il progetto acquisì ulteriore vigore con il successivo arrivo di Oracle e con l'annuncio che Monterey sarebbe risultato compatibile con Linux, così come la prossima versione di AIX.

Nel frattempo IBM aveva rilevato l'area semiconduttori di Motorola e poteva costruire in proprio i processori PowerPC G3, G4 e G5, che le permettevano altresì di evolvere i sistemi RS/6000 e RISC. Spariva così la necessità di portare AIX su Intel.

SCO digerì malissimo il rinnovato impegno di Big Blue nella realizzazione di microprocessori, ma soprattutto questa produttiva relazione tra Sequent e IBM. Molte delle successive accuse contro quest'ultima ebbero per oggetto proprio le tecnologie Sequent NUMA acquisite da IBM.

Ancora una volta va dunque sottolineato come, nell'attacco a testa bassa contro il pinguino, SCO tenti di accreditare a Linux la colpa delle pessime relazioni con IBM, mentre in realtà queste risalivano ad un periodo ben precedente.

Il colpo di mano degli ex-SCO

Ransom Love non era in grado di dettar legge nella propria azienda. Ne era sì il presidente, ma il business della vecchia Caldera non contribuiva ai ricavi della società, provenienti interamente dalla sezione scorporata da SCO. Tra questi primeggiava quell'OpenServer snobbato dallo stesso Love e i cui introiti continuavano a finire nelle casse di Tarantella. Per quanto il ricavato non riuscisse comunque a coprire i costi dell'azienda (si viaggiava con milioni di dollari di perdite per ogni trimestre fiscale), costituiva però l'unica fonte di guadagni. E in nessuna azienda il dirigente che ottiene delle entrate può vedersi subalterno ad altri che invece non contribuiscono affatto.

Dopo anni di frustrazione dovuti alle intimidazioni di Doug Michels, finalmente i manager di SCO si sentivano liberi di seguire le proprie idee, indipendentemente da quelle che potevano essere le linee guida di un leader debole come Ransom Love. Per anni SCO aveva seguito Microsoft nella condanna del software libero come male assoluto dell'universo informatico. Poi d'un tratto s'era mostrata amichevole, ma una volta sperperate le risorse acquisite chiedendo ai risparmiatori soldi

‘per Linux’, assumerà nuovamente atteggiamenti contrari – fino, appunto, all’attacco legale.

Nel frattempo, dopo soli pochi mesi di vita la nuova Caldera era già di fronte alla peggiore crisi aziendale. Il finanziamento di Novell, Sun, e Citrix, i soldi degli azionisti con l’IPO e, non da ultimo, la vasta rete di clienti SCO non permettevano a Caldera di prendersi del tempo per tentare vanamente di mettere in piedi un efficace business model.

In prospettiva, inoltre, l’abbandono degli sviluppi su IA-64 da parte di IBM rendeva improbabile una evoluzione per gli Unix SCO.

In un estremo tentativo di riacquistare potere all’interno del proprio gruppo aziendale, Ransom Love tentò di monetizzare l’area Linux, proponendo anche per le proprie distribuzioni una speciale licenza ‘per seat’, ampiamente in contrasto con le licenze per il rilascio di Linux. La comunità open source insorse: a nessuno è permesso aggiungere termini di licenza in contrasto con la licenza base onde limitarne la distribuzione. Caldera fu costretta a fare rapidamente marcia indietro, passando ad uno schema più soffice indirizzato ai soli utenti commerciali, per poi rimangiarselo del tutto. Ransom Love perse così ulteriore credibilità all’interno della propria organizzazione. Mr. Yarro decise allora di sbarazzarsene.

La presenza di due marchi era senza dubbio fuorviante. Il business che manteneva viva Caldera non aveva nulla in comune con il mondo open source, e il denaro che aveva permesso a SCO di sopravvivere ad una morte annunciata era stato ottenuto nel nome di Linux. Il mercato non rispondeva affatto bene e nel frattempo era anche crollata l’euforia degli investimenti nella new economy. Le aziende più esposte, come Caldera,

dovevano trovare dei metodi per ridurre sensibilmente i costi e dare l'impressione che tutto fosse in evoluzione, così da attivare un nuovo giro di finanziamenti.

Era il momento di tirare la cinghia. Il 6 settembre, a pochi mesi dalla conclusione dell'acquisto di SCO, Caldera mandò a casa alcuni manager e 51 sviluppatori, in aggiunta ai 190 già licenziati in maggio da SCO prima della fusione. La nuova "Scaldera", come veniva ufficiosamente chiamata allora, chiuse l'anno con perdite operative consistenti, un management rabberciato e un settore tecnico ridotto di oltre il 30 per cento. Dopo Ericsson, Siemens, Motorola, HP e SGI – aziende di ben altro calibro – Caldera risultò una delle aziende che operò i più decisi tagli d'organico nel corso del 2001.

Oltre a quello di veterani OS/2 e Java come Nick Petreley, suscitarono molto clamore i licenziamenti mirati degli sviluppatori più conosciuti nella comunità Linux, tra cui quel Juergen Kienhoefer a cui si dovevano le oltre 40.000 linee di codice necessarie a realizzare la Linux Kernel Personality, un sistema che permetteva agli Unix proprietari di SCO di eseguire senza modifiche le applicazioni Linux, avvantaggiandosi del grande parco software in circolazione.

Evidentemente Kienhoefer e gli altri avevano assunto posizioni un po' troppo intransigenti sul rispetto delle licenze e sull'effettiva proprietà del codice disponibile in SCO, spesso riscritto usando come riferimento il codice del kernel Linux o degli altri programmi disponibili pubblicamente.

Persino il vicepresidente Jim Wilt, padre-padrone del canale di vendita SCO, 'scompare' dalla gestione aziendale e non troverà nulla di meglio da fare che comprarsi un dominio web a basso costo su cui inserire il proprio curriculum per offrire servizi pro-

fessionali di marketing, sviluppo aziendale e negoziato.¹ Il comportamento spregiudicato di Caldera nell'imposizione di licenze anti-GPL, ma forse ancor di più il supporto 'esterno' che Love offrì a Craig Mundie, tra i manager di prima linea di Microsoft, sostenendo l'idea che sarebbe stato impossibile fare affari con una licenza come la GPL, fece scattare persino Richard Stallman, il quale si premurò di censurare amaramente il comportamento della nuova Caldera. Da Mundie sarebbe nata poco dopo la strategia di Microsoft per contrastare l'utilizzo della GPL tramite la propria licenza Shared Source, che permette a clienti particolarmente blasonati di visualizzare il codice sorgente dei programmi Microsoft, senza però poterlo utilizzare in alcun modo.

In seguito alla ristrutturazione e alle feroci critiche della comunità, Ransom Love svanirà da ogni attività pubblica per dedicarsi alla realizzazione di una iniziativa comune con altre tre importanti firme Linux: la brasiliana Conectiva, la tedesca SuSE, leader del mercato europeo, e Turbolinux, che vantava buona visibilità in Cina e Giappone. Il consorzio, battezzato UnitedLinux, prendeva spunto dai nuovi standard Linux appena pubblicati e non ancora accettati dal leader di mercato RedHat, che non di rado preferiva procedere a modo proprio, e che quindi avrebbe accettato con scarso entusiasmo i limiti posti dalla Linux Standard Base.

Si trattava di una motivazione chiaramente speciosa, ma la mossa servì a Caldera per liberarsi di due ulteriori grossi gruppi di sviluppatori che aveva assoldato in California e in Germania, i quali passarono così sotto il diretto controllo di SuSE, tradizionalmente caratterizzata da grandi capacità di sviluppo software. La nuova società perse il più consistente gruppo di sviluppatori Linux, e si trovò con quattro quinti del management prove-

¹ <http://www.lateralgroup.com>

niente dalle vecchie fila di SCO. Molti di questi avevano scavalcato gli stessi ex-dirigenti, silurati per poca fedeltà alle nuove idee di evoluzione, e certamente vantavano conoscenze minime sulle dinamiche tipiche dell'open source.

Dopo aver reso pubblicamente inutilizzabili le proprie tecnologie Unix per acquisire un posto di rilievo nel campo dei fornitori di servizi basati su Linux, SCO non poteva far altro che richiudersi nel vecchio business di una volta: gli Unix proprietari.

All'interno di UnitedLinux, l'unico contributo di Caldera al software open source, Lizard, sarà rapidamente sorpassato a livello tecnico dallo YaST (Yet another Setup Tool) di SuSE, dotato di qualità e potenzialità nettamente superiori. Caldera armeggiò allora per presentarsi come fornitore di servizi su sistemi operativi professionali di alto livello basati principalmente sul marchio UnixWare, modificato poi in OpenUnix, sperando di portare quel tanto di confusione nei clienti della distribuzione libera OpenLinux di Caldera, o dello Unix a basso costo OpenServer, da indurli a migrare verso il proprietario e costoso OpenUnix/UnixWare. Nel frattempo, poiché il valore delle azioni fluttuava in caduta libera, l'azienda dovette ricorrere ad una ricombinazione uno a quattro dei titoli onde rimanere al di sopra del minimo di un dollaro imposto dalle regole del Nasdaq. Al contempo rilevò completamente i diritti di OpenServer (lo SCO Unix, ex-Xenix), rimaste in mano a Tarantella Inc., e riacquistò a prezzo di realizzo – appena 93 centesimi di dollaro per azione – le quote tuttora in possesso di Tarantella Inc. e di MTI Technology, circa il 33 per cento dell'intero patrimonio azionario emesso allo scopo di ridurre l'indebitamento complessivo.

Ransom Love venne messo decisamente da parte: il suo nome era troppo legato ai ripetuti fallimenti di Caldera Linux per

poter essere giocato in un'azienda che doveva risollevarsi da una crisi mortale. Dedicatosi inizialmente al progetto UnitedLinux, finì per abbandonare del tutto la scena Linux e l'ala protettiva di Noorda. Rientrerà più tardi in Progeny, un distributore Linux di Debian, per evidenziarsi come uno dei più acrimoniosi accusatori delle nuove strategie di The SCO Group Inc. Caldera aveva bisogno di un capitano nuovo di zecca, anche perché nel consiglio di amministrazione ormai dei sostenitori di Linux non esisteva più traccia.

Darl tutto d'un pezzo

Darl McBride è il prototipo di quello che gli hacker definirebbero 'tie', un uomo in cravatta, anzi solo cravatta. Mai transitato da una struttura tecnica, McBride sembra un venditore di prima classe. Ciò nonostante, dopo averlo conosciuto, ben pochi comprenderebbero da lui una macchina usata.

Quando sali in sella a Caldera, Darl operava come dirigente del sito Internet per la gestione del software di pianificazione presso la Franklin Convey. A rileggerla oggi, la dichiarazione del fondatore Stephen Convey, nel comunicato stampa in cui si saluta la partenza di McBride, comunicandone l'adesione a Caldera, ha tutto il sapore di una velata presa in giro: "I clienti Caldera, i partner, gli sviluppatori e gli impiegati, e tutti gli azionisti apprezzeranno e trarranno benefici dalla leadership che McBride porta nell'azienda". Un po' come dire: "Peggio per voi. Mors tua vita mea".

L'addio di Darl McBride arriva infatti dopo appena due anni di attività, periodo in cui la Convey subisce perdite per ben dieci milioni di dollari. Il sito da lui gestito verrà definitivamente chiuso. Sembra che McBride fosse passato alla Franklin Convey per realizzare, da solo, quello che nella precedente società,

la Pointserve, non era riuscito a fare pur avvalendosi di un gruppo esperto e di maggiori finanziamenti. Questo il giudizio espresso dall'interno della stessa Pointserve:

“Per un breve periodo il CEO di SCO, McBride, è stato CEO di Pointserve, dove ha fatto fallire un buon modello aziendale, pagato un sacco di dollari per una azienda esterna di consulenza per tirar fuori un nuovo modello aziendale balordo basato su pubblicità Internet fatta con finestre pop-up. Ha licenziato la maggior parte dello staff professionale, e poi è stato a sua volta licenziato. La società è rimasta con programmatori demoralizzati e senza più un prodotto da vendere. Ho lavorato per Pointserve e ho parlato con lui per circa un'ora, è tutto fumo e specchi, e parole senza senso. Mi dispiace SCO, ma ragazzi avreste dovuto controllare prima!”².

Né si può dire che gli fosse andata meglio in precedenza. Un'altra start-up, di cui assunse la guida come amministratore, lo estromise subito dopo aver ricevuto un cospicuo finanziamento, accusandolo di voler perseguire progetti personali e di aver inutilmente cambiato nome all'azienda, da SolutionBank a SBI. Per non parlare della IKON, dove resistette solo l'arco di un anno, a cavallo tra il 1996-97, pur riuscendo a trascinare il proprio datore di lavoro in una rabbiosa causa civile dove ottenne un ambito premio di produzione che gli era stato contestato – non avendo evidentemente raggiunto gli obiettivi di vendita prefissati, ed essendo stato allontanato dagli uffici dello Utah. Analoghi gli insuccessi accumulati precedentemente in Novell: nell'era Frankenberg era stato prima spedito in Giappone per salvare gli ultimi rimasugli della rete di vendita ormai al col-

² Dan Pani, <http://www.sys-con.com/author?id=2655>

lasso, e poi gli era stata affidata la NEST, il piccolo nido in cui NetWare avrebbe dovuto essere eseguito nei sistemi embedded, di cui la storia non ricorderà altro che qualche comunicato stampa e un inglorioso flop.

Arrivato dunque in Caldera/SCO con 18 anni di “executive management” alle spalle, come recita la biografia ufficiale, a ben vedere McBride aveva frequentato solo realtà economiche ridotte o appena avviate, ma che comunque non seppe guidare per più di 24 mesi, oppure sotto-unità operative limitate e già sulla via del declino. Si poteva forse considerare il capitano adatto per guidare la nuova SCO?

Di sicuro era mormone, aveva prestato servizio in Giappone come predicatore e, al colmo della esaltazione personalistica del fratello Kevin, si diceva che in gioventù avesse persino fatto volontariato insegnando informatica al proprio professore di inglese. Mentre pare che il padre, dopo avergli assegnato una vacca e un campo di grano, costrinse il mitico Darl a sbrogliarsela da solo. Ransom Love, sollevato dall’incarico e messosi a cincischiare con UnitedLinux, considerava rude e scostante la personalità di McBride. Eppure lo aveva chiamato in azienda in nome della lontana amicizia nata in Novell, della comune frequentazione della Chiesa dei Santi degli Ultimi Giorni, (soprattutto) dei consigli di Ralph Yarro e del placet di Ray Noorda. Tuttavia Joe Firmage, VicePresidente di Novell Service Group e grande amico di Darl ai tempi del NEST, non aveva dubbi. Con una lettera pubblica lo sconfessò alla grande, avviando di fatto la battaglia che porterà poi Novell a schierarsi a favore di Linux.

Una mossa quest’ultima che McBride non avrebbe mai sospettato. Oltre ad essere i principali investitori in SCO e a mantenere ottimi rapporti con Caldera, gli “uomini Novell” si con-

sideravano tutti amici strettissimi e, non ultimo, legati anche da vincoli familiari con Noorda e i suoi accoliti. Mormoni anch'essi e fratelli. Ma evidentemente non fu sufficiente. Quando Darl McBride prese in mano le redini di Caldera, il 27 giugno del 2002, ricevette istruzioni precise su cosa fare.

La strategia comunicativa di McBride

Nell'ultimo anno, Ransom Love in Caldera aveva avuto la costante preoccupazione di promuovere lo spostamento della clientela Unix verso Linux, o anche viceversa, secondo quale delle due piattaforme risultasse maggiormente adeguata alle esigenze dello specifico cliente. In realtà si trattava di un'operazione di fatto impraticabile. Unix prevedeva vendite di mere licenze, che significava ricavi per la società senza dar nulla in cambio agli utenti se non il diritto di usare qualcosa. Linux invece legava i propri introiti alla fornitura di veri e propri servizi, tra cui installazione e manutenzione. Mancando l'integrazione dei diversi business model, era impossibile pensare di far migrare un utente da uno schema all'altro senza che qualcuno non ci rimettesse. Nel caso del passaggio da Linux a Unix, il cliente avrebbe speso parecchio per ottenere la licenza e ancor di più per garantirsi il supporto tecnico. Nella direzione opposta, era Caldera a dover rinunciare alla comoda rendita della licenza, per offrire un servizio sotto costo di quanto veniva già offerto in Unix. Ciò nonostante l'idea di Ransom Love affascinò McBride: bisognava identificare i clienti che usavano Linux per imporre loro una tariffa supplementare. Come e quale?

Nel frattempo McBride cercò anche di posizionare Caldera al centro della competizione in ambiente Linux fronteggiando il principale rivale RedHat. Si trattava di due aziende sostanzial-

mente comparabili, non per la capitalizzazione in borsa – dove RedHat viaggiava ancora nel firmamento e Caldera era praticamente ridotta a una nullità – ma per il livello delle entrate. Con la differenza che tutti i guadagni della prima derivavano dal modello open source, mentre la seconda aveva un flusso finanziario, anche se minore, dovuto alle vecchie licenze proprietarie. Non a caso nel secondo giro di finanziamenti Caldera presenterà dati finanziari comparabili a quelli di RedHat, onde sfruttarne la fortuna, pur se indirettamente.

Il tradizionale Forum annuale dei clienti SCO, l'anno precedente ribattezzato Caldera Forum, viene intanto rinominato Caldera/SCO Forum dallo stesso McBride, a pochi giorni dall'investitura ufficiale. Sul palco il nuovo amministratore e presidente si presentò indossando giacca di pelle, pantaloni neri e occhiali a specchio avvolgenti, per raccontare ad un pubblico per metà divertito e per metà incredulo, che quello che rende grande la mitica motocicletta Harley-Davidson è l'immortale patrimonio accumulato. Il messaggio, per nulla velato, è che allo stesso modo avrebbe dovuto comportarsi Caldera/SCO sotto la sua guida. Affrontando la questione Linux, l'intervento di McBride raggiunse l'apice: dopo essersi versato un bicchiere di acqua riproponendosi di venderlo, eccolo mostrare, con la teatralità dell'imbonitore televisivo, una graziosa bottiglietta di acqua minerale di marca e decantarne l'alto prezzo di mercato. Sarebbe bastato, a suo dire, il nome SCO Linux, "powered by United-Linux", sui sistemi certificati IBM o HP, per far vendere anche il 'gratuito' Linux.

McBride uscì dal Caldera/SCO Forum avendo già deciso il nuovo nome della società, e rapidamente la holding Caldera International divenne The SCO Group Inc. – pur se al momento sol-

tanto nella comunicazione pubblica: si presentava come “The SCO Group, a Caldera International Inc. company”. In quest’ambito SCO significa semplicemente SCO, e non è l’acronimo di Santa Cruz Operations. D’altronde la nuova SCO non aveva proprio nulla in comune con la società di Santa Cruz, se non un piccolo gruppo di sviluppatori ‘fedeli alla nuova linea’, e un pugno di manager. Con sede legale a Lindon nello Utah, presso gli uffici del Canopy Group di Noorda.

Per la terza volta Caldera era passata a miglior vita. Stavolta in maniera definitiva. E il contraccolpo si rivelò notevole.

Rob Hicks era rientrato in Novell dopo il collasso di Novonyx e Bryan Sparks aveva perduto la sua Caldera Thin Clients Inc., venduta da Noorda alla Motorola con il nome di Lineo, per poi rifugiarsi in DeviceLogics Inc., dove sviluppava sistemi embedded basati su DOS. Ransom Love aveva perso la sua Caldera Systems Inc. per realizzare The SCO Group Inc. Il “Corsair Project”, dopo aver concluso praticamente nulla, era stato disperso tra l’indifferenza generale.

Ora esisteva “The SCO Group Inc.”, basato su un progetto di rinascita che voleva mimare la Harley-Davidson e il tentativo di piazzare un’etichetta su Linux per poterlo vendere al meglio. McBride sapeva che citando la celebre Harley-Davidson avrebbe richiamato alla mente di ogni statunitense la grande crisi che colpì quest’azienda dopo la cessione a favore della AMF e fino a quando, negli anni ‘80, lo stesso management interno, grazie all’investimento di alcuni discendenti della vecchia famiglia Harley, la rilevarono per tentarne un estremo, critico salvataggio. In quel caso, però, il team aziendale centrò le speranze di rinascita sulla possibilità di generare ricavi dallo storico marchio, sul drastico miglioramento dei maggiori prodotti e sulla maggiore

attenzione riservata ai canali di vendita. Esattamente i semplici concetti elencati dallo stesso McBride, ma alquanto difficili da mettere in atto nel contesto informatico dei nostri giorni.

Il rilancio del marchio, riprendendo l'esempio della Harley-Davidson, fu paradossalmente affidato non tanto alle motociclette, quanto ad una azzeccata scelta della licenza per il lancio di una linea esclusiva di capi d'abbigliamento e accessori. The SCO Group non annoverava una simile opportunità, potendo solo approfittare delle royalty relative all'unico prodotto in grado di generarne: lo Unix AT&T, acquistato dalla Novell di Ray Noorda per oltre un miliardo di dollari e finito nella scuderia Caldera per pochi milioni di dollari.

“Teoricamente – ribadì McBride al Caldera/SCO Forum, in una chiosa che rimarrà centrale nel corso dell'intera vicenda – in questa proprietà dovrebbe esistere un qualche valore, compreso tra un milione e un miliardo di dollari. Voglio scoprire qual è la proprietà intellettuale concreta e tangibile che quest'azienda possiede”.

A meno di tre settimane dalla salita a CEO, McBride diede mandato agli avvocati di trovare, nella vasta mole di contratti in circolazione, il bandolo di questa matassa. “Noi possediamo i sorgenti di Unix – affermò Opinder Bawa, neo-assunto vicepresidente della divisione tecnologica – e quindi abbiamo il diritto di riscuotere i pagamenti per le licenze”. Lo stesso McBride corse ad aggiungere: “Ovviamente Linux deve molto a UNIX, ma non ne usa il codice. Noi non pretenderemo mai nulla”. La prima affermazione pubblica destinata ad essere rimangiata.

Caldera/SCO, intanto, riscuoteva già oltre seicentomila dollari in licenze Unix ricontrattate con i licenziatari, ma il nucleo della nuova offerta SCO avrebbe dovuto essere un nuovo servizio

denominato “SCObiz”, in partnership con Vista.com e l’annessa piattaforma di e-commerce. Tale servizio prevedeva la generazione da parte di SCO dei profili aziendali della clientela, rispetto ai quali Vista.com si sarebbe posto come mediatore di servizi utili alle esigenze dei singoli clienti, compresi eventuali nuovi servizi o prodotti SCO. Un tipo di offerta che peraltro sembrava essere invocata a gran voce dagli stessi rivenditori SCO, i quali avrebbero così trovato modo di riscuotere piccole royalties in cambio dell’identificazione delle specifiche attività dei clienti. Pur essendo lanciato in modo sfarzoso e certamente sovradimensionato (vedi lo slogan “Born to raise sales”), il progetto SCObiz mancherà di dare frutti tangibili per l’intero anno successivo. In realtà si trattò di un sistema di intermediazione di e-commerce piuttosto comune e solitamente fornito con costi di gran lunga inferiori a quelli richiesti da SCO. L’iniziativa, inoltre, non ottenne neppure un proprio sito, essendo Sco-biz.com già acquisito, e fu quindi costretta a ripiegare in una sottopagina del sito aziendale.

Nasce ufficialmente The SCO Group Inc.

Il 2 settembre 2002 Caldera International Inc. annunciò pubblicamente il cambio dell’ intestazione sociale in “The SCO Group Inc.”. Da quel momento l’azienda si presenterà sempre con il nuovo nome, mentre la dicitura Caldera rimarrà prima riservata alla sola distribuzione Linux, e poi verrà completamente dimenticata quando anche Caldera Linux diventerà SCO Linux.

Il cambiamento provocò una fervida attività di comunicazione: prima l’avviso per l’imminente uscita della prima versione di UnitedLinux, poi l’intervento di McBride al Commonwealth

Business Forum sulla modernizzazione attraverso l'open source, in cui si lancerà con enfasi in una accorata presentazione dei prodotti open source di Caldera. Il primo ottobre fu anche resa pubblica l'assegnazione della certificazione Linux Standard Base a OpenLinux, per primo insieme a Mandrake Linux. Ancora, il rilascio dell'Insight Connector per consentire ai programmi di posta elettronica di usare client Microsoft. Da notare come tutte queste attività di SCO coinvolgano aziende in un modo o nell'altro legate alla rete di relazioni amicali o finanziarie di Ray Noorda, Ralph Yarro o Darl McBride. Verranno 'sparati' comunicati stampa su incredibili accordi tra SCO e realtà aziendali che, a ben vedere, sono sempre tutte riconducibili ad amicizie personali di McBride (come nel caso della 'one-man' azienda TSG di George J. Fulop, o la Nhosa in Giappone), oppure ai finanziamenti del Canopy Group di Noorda (come la Center 7). Proprio la Center 7 guidata dal Canopy Group, come Caldera/SCO, sarà al centro di una ulteriore causa legale, questa volta contro la Computer Associates, da cui ricaverà altri 40 milioni di dollari.

A dicembre SCO diffuse intanto una ulteriore versione di UnixWare, che non avrà significativi miglioramenti se non l'aggiunta del sistema automatico di aggiornamento denominato SCO Update Service, pedina importante nella strategia di eliminazione degli intermediari del canale distributivo. Pur di raggranellare una ulteriore percentuale sul prezzo di vendita, SCO decise infatti di avviare tale strategia onde qualificare le società locali come fornitori diretti dei clienti SCO, in aperta concorrenza con quei rivenditori che pure avevano portato al successo l'azienda nell'era dei Michels, e che spesso erano stati allevati dalle amichevoli relazioni personali con Jim Wilt. L'arro-

ganza di McBride si riverserà in molte realtà nazionali, specie in quelle europee, dove non raramente saranno tagliate di netto, da un giorno all'altro, tutte le relazioni con rivenditori e distributori, tanto da portare in alcuni casi a vere e proprie cause legali, come in Olanda. Parecchi rivenditori, peraltro, non erano certo rimasti in attesa delle determinazioni aziendali, e avevano colto l'occasione della prima offerta di Caldera OpenLinux per iniziare quelle migrazioni che verranno seguite, durante il breve periodo legato all'avvento di UnitedLinux, da più ampie operazioni di distacco a favore della solida distribuzione tedesca SuSE, di qualità certamente comparabile a UnixWare e superiore a SCO Unix.

Comunque sia, The SCO Group avocò di fatto a sé gli introiti degli sfortunati clienti incastrati nei sistemi SCO, lasciando spesso ai canali in loco soltanto le briciole del supporto tecnico. In Europa, poi, queste carenze apparivano ancora più evidenti per l'ulteriore mancanza di strutture tecniche di assistenza e di laboratori dove scalare i problemi, non di rado rilevanti, riguardo il funzionamento degli Unix proprietari. Ciò perché i laboratori europei erano stati ereditati da Tarantella e non effettuavano supporto per Unix, mentre al contempo parecchi sviluppatori già allocati sui progetti SCO erano passati a SuSE. Il 28 ottobre 2002, Chris Sontag assumerà il ruolo di vice presidente per i sistemi operativi. Amico di vecchia data, proveniva da una start-up creata dopo essere fuoriuscito da Novell e si vedrà affidato il delicato compito di fronteggiare il problema delle licenze Unix. Anche Sontag, al pari di McBride e Yarro, è figlio dello Utah e venne spedito, appena superata la maggiore età, come predicatore mormone in giro per il mondo. Anche Sontag, come McBride, usò la propria posizione in Novell per

realizzare un progetto imprenditoriale individuale, senza molti sbocchi, dopo aver razzolato, sempre come McBride, i soldi dei risparmiatori e delle agenzie d'investimento. E come McBride, Sontag non aveva granché da perdere saltando sul carro di The SCO Group: poteva invece guadagnare parecchio dimostrando fedeltà all'amicizia con Yarro e con le società di investimento che ruotavano intorno a Ray Noorda.

SCO scalda i muscoli per l'attacco legale

I primi giorni dell'anno 2003 portarono un inatteso successo all'iniziativa di Ransom Love: UnitedLinux raccolse le adesioni di IBM, HP e AMD. Mentre SCO stava evidentemente pensando ad altro, SuSE (partner-chiave di UnitedLinux) mise a buon frutto gli ottimi rapporti che era riuscita a crearsi negli ultimi anni di duro lavoro.

Nelle stesse settimane, sul fronte SCO le ricerche del team legale – e di conseguenza le pubbliche dichiarazioni di McBride, Bawa e Sontag – si fecero sempre più circostanziate circa l'opportunità da parte della società di avviare azioni legali contro i licenziatari di Unix non in regola con gli obblighi previsti. L'iniziativa, pur non essendo dichiaratamente ostile nei confronti di Linux (che rimane prodotto di punta di SCO), provocò una ridda di voci contrastanti.

Il 10 gennaio, la webzine *Client Server News* diffuse la notizia secondo cui SCO stava pensando di assumere il quotato avvocato David Boies per portare avanti l'accusa contro Linux e pretendere così una licenza da tutti gli utenti del pinguino. Tre giorni dopo, Tina Gasperson su *NewsForge*, una delle webzine più note del mondo open source, riportò di aver contattato lo studio legale di Boies: viene categoricamente smentito che qualcu-

no tra gli oltre 100 legali dello studio stia lavorando a questioni concernenti i diritti di proprietà intellettuale di SCO. D'altronde, fa notare la segretaria personale, né Mr. Boies né altri avvocati dello studio si sono mai occupati di un simile settore. Lo stesso McBride si trova costretto a dover emettere un comunicato stampa di smentita (che poi sparirà dal sito SCO, continuamente 'corretto' come più conviene all'azienda):

"[...] Non abbiamo raggiunto alcuna decisione definitiva su qualche azione da fare [...] Come parte normale del nostro lavoro, SCO ha avuto alcune discussioni con alcuni esperti legali nel campo della proprietà intellettuale, e queste consultazioni hanno incluso David Boies. Al contrario di quanto riportato in *Client Server News*, SCO non ha ingaggiato Mr. Boies per prendere alcuna contromisura legale contro i nostri rivenditori Linux".

Il 22 gennaio 2003 nella cornice del LinuxWorld, la maggiore fiera annuale dedicata al pianeta Linux, The SCO Group annuncia la creazione dell'ufficio SCGroup, guidato da Chris Sontag, e la disponibilità di SCO System V for Linux a 149 dollari per CPU – una strana libreria di 'compatibilità' tra Linux e Unix, per "coprire con una licenza la proprietà intellettuale Unix in Linux". Nella stessa occasione viene presentato un accordo ufficiale con lo studio legale Boies, Schiller e Flexner per la tutela dei diritti intellettuali di SCO. Viene così svelato il 'segreto' su come e quando far pagare ad ogni utente Linux una piccola somma, dietro la minaccia di una causa da parte del più rinomato avvocato del momento. David Boies è alquanto noto, anche al pubblico Linux, per aver rappresentato il grande accusatore di Microsoft nel procedimento Antitrust fomentato da Ray Noorda. Boies ha anche rappresentato Al Gore nella conta dei voti per l'elezione come presidente degli

Stati Uniti persa dai democratici per una manciata di voti (nel 2000, in Florida).

Il punto di vista di SCO e McBride – o quasi

Il 6 marzo 2003 Caldera International Inc. (la ex-Caldera Holding incorporata in Delaware) avvia la causa civile contro IBM presso il Tribunale di Stato dello Utah in Salt Lake City. Chiedendo prima uno, poi tre miliardi di dollari (successivamente saliti a cinque) come riparazione dei danni subiti, Caldera/SCO sostiene che IBM, tramite il suo supporto allo sviluppo di Linux, ha violato i contratti firmati dalla stessa IBM con i predecessori di Caldera/SCO sulla proprietà di Unix relativamente alla non-divulgazione del codice Unix.

Questo l'assunto-base di una saga che, a oltre un anno dal suo avvio, continua a provocare contraccolpi a tutto tondo nell'intero mondo informatico e in quello open source/software libero in particolare. Superato un certo e inevitabile stupore generale, è quindi il caso di chiedersi: ma è proprio vero che SCO abbia torto marcio su tutto il fronte? Al di là delle certosine diatribe legali, SCO potrebbe star mettendo il dito nella piaga di Linux? E Darl McBride potrebbe anche avere ragioni da vendere, no?

È allora il caso di osservare più da vicino il punto di vista di SCO – e del suo boss – per come si è estrinsecato in quest'arco di tempo, anche e soprattutto al di fuori dell'istanza civile originaria. La quale, vale la pena di ribadirlo, è centrata sulla convinzione che IBM abbia introdotto impropriamente codice e metodi Unix all'interno di Linux. Nello specifico, IBM viene accusata di appropriazione indebita di segreti commerciali (poi derubricata), interferenze lesive, competizione scorretta e rottura di contratto; il tutto senza però identificare pubblica-

mente il codice imputato e mostrandone solo una minima parte ad alcuni esperti e giornalisti sotto la tutela del tipico accordo di non divulgazione (non-disclosure agreement).

Analogo il tono della successiva nota di inizio maggio, in cui SCO replica alla formale difesa depositata poco prima da IBM, dove viene negato sostanzialmente ogni addebito. Sottolineando la mancanza di dettagli in replica alle motivazioni specifiche addotte da SCO, questa annuncia che dal 13 giugno revocherà la licenza AIX di IBM. La ragione è che tale licenza può sì considerarsi perpetua e irrevocabile (come sostiene Big Blue), ma soltanto nel caso in cui l'azienda che ne è titolare "si attenga alle norme della propria licenza Unix e non si appropri illegalmente del codice sorgente Unix di proprietà SCO". Nel documento quest'ultima si dice "incoraggiata dalla replica di IBM", ribadendo di possedere "chiare prove" in appoggio alle accuse avanzate finora, nonché "forti testimonianze che si stanno mettendo in fila a sostegno del nostro caso". Prove e testimonianze, conclude il documento, che "stanno per approdare in aula". Non sarà proprio così, anche perché, nel frattempo, il castello di accuse verrà smontato pezzo per pezzo grazie soprattutto all'analisi dell'Open Source Initiative e di Eben Moglen, consigliere della Free Software Foundation (si veda il capitolo successivo). Motivo per cui in estate Caldera/SCO dichiara di voler insistere soltanto sui diritti contrattuali, pur riservandosi la facoltà di perseguire in futuro eventuali infrazioni al copyright o ad altri accordi. Nel frattempo, a luglio l'azienda annuncia l'imminente lancio di un programma di licenze per gli utenti Linux, senza rivelarne ancora i dettagli. In autunno l'ulteriore e dettagliata stesura legale di SCO contro la replica punto per punto di IBM si chiude in maniera perentoria: viene chiesto il completo

rigetto di tale replica. In alternativa, invoca una sentenza comunque a proprio favore, imponendo allo sconfitto il rimborso delle spese processuali, ma tornando a insistere per un vero e proprio dibattimento in aula con tanto di giuria. Ciò non senza aver presentato ulteriori denunce formali, stavolta contro qualche imprenditore 'reo' di usare Linux. Dopo aver diffuso a novembre un ultimatum di 90 giorni contro gli utenti commerciali del pinguino, a inizio marzo 2004 lo SCO Group querela il maggior rivenditore statunitense di accessori e ricambi auto, AutoZone, e il noto gruppo automobilistico Daimler Chrysler. Motivo? Al solito, violazione del copyright su Unix.

Restano così dure e risolte le posizioni di SCO, pur se annacquate giocoforza nel corso del tempo, e permane l'attesa per un imminente sbocco giudiziario, onde superare lo stallo dovuto all'acquisizione del materiale probatorio che va avanti da troppi mesi. Non a caso, all'inizio del marzo 2004, la corte distrettuale dello Utah ha ordinato ad entrambe le parti in causa di esibire tutte le prove atte a chiarire una volta per tutte la questione. In particolare, stavolta sembra proprio che SCO dovrà mostrare (entro 45 giorni) tutte le linee di codice che asserisce essere state copiate in Linux dai sistemi operativi UNIX di IBM. Il giudice ha anche richiesto alla ex-Caldera di identificare e presentare l'intera porzione di sorgente Linux su cui reclama i diritti e di fare lo stesso per quel codice che ha distribuito a terze parti.

GNU, GPL e la comunità open source sono dei gran cattivi
Ma l'iter giudiziario e la pila di documenti in stretto legalese³ non bastano certo a spiegare quest'attacco frontale – odio svi-

³ Per la raccolta completa dei documenti legali originali sul caso SCO-IBM si veda:

<http://www.sco.com/ibmlawsuit/>

Sito cruciale per commenti, letture ragionate e aggiornamenti sugli aspetti legali dell'intera faccenda: <http://www.groklaw.net>

scerato secondo alcuni – lanciato da SCO contro il pianeta Linux/open source/software libero. Forse le motivazioni ad ampio raggio vanno cercate piuttosto in lettere aperte e altri testi più informali diffusi nel frattempo, partendo dal documento più recente firmato da Darl McBride: la missiva inviata ai parlamentari statunitensi in data 8 gennaio 2004. Diffusa online dall’Open Source & Industry Alliance (OSAIA) con base a Washington, DC, la lettera⁴ si concentra sostanzialmente su una questione specifica: il software open source, soprattutto quello diffuso sotto la licenza GNU/GPL, rappresenta un’impellente minaccia per l’economia e per la sicurezza nazionale, mettendo in discussione il ruolo di leadership statunitense nell’innovazione tecnologica globale.

Illustrando significato e procedure operative del software open source e del “prodotto open source più ampiamente usato”, ovvero Linux, McBride scrive che questi “vengono sviluppati e distribuiti (spesso a costo zero) sotto uno schema chiamato GNU General Public License (GPL) che, sostengono alcuni, si pone in diretta contraddizione con le leggi USA sul copyright, con il Digital Millennium Copyright Act e con la recente sentenza della Corte Suprema nel caso *Eldred v. Ashcroft*”. Per maggior chiarezza, specifica ancora la lettera, un allegato affronta in dettaglio i problemi della GPL e “le modalità secondo cui questa viola le attuali disposizioni statunitensi”. L’affondo si allarga fino a colpire Richard Stallman in persona, pur senza nominarlo, mentre poco oltre si legge che “ogni installazione open source sostituisce o impedisce le vendite di software proprietario, tutelato dalle relative licenze e copyright. Ciò

⁴ http://www.osaia.org/letters/sco_hill.pdf

significa meno posti di lavoro, meno incassi e meno incentivi all'innovazione per le software-house”.

La bandiera del nazionalismo corporativo viene poi ripresa da McBride quando spiega che, al posto di Unix o Windows, “i governi di ogni parte dell'Europa e dell'Asia stanno usando Linux, spesso scaricato gratuitamente da Internet. Fatto particolarmente irritante visto che Linux contiene migliaia di linee di codice del mio UNIX proprietario, linee di codice per le quali non ho ricevuto in cambio nessuna remunerazione”. Una serie di accuse pesanti, non certo nuove, ma stavolta dettagliate al massimo onde affermarsi nello specifico contesto parlamentare. A cui replica, con toni altrettanto decisi, la stessa OSAIA⁵ con una nota in cui si sottolinea tra l'altro il proseguimento della campagna di FUD (Fear, Uncertainty, Doubt) già avviata tramite lettere e denunce passate. Punto centrale è che le presunte minacce segnalate ai legislatori dal boss di SCO “sono tutte asserzioni oltraggiose e false”. Secondo Ed Black, presidente del consorzio industriale open source, “il software scelto da migliaia fra le più grandi aziende nazionali, non solo in ambito high-tech, non costituisce una minaccia, eccetto che per coloro che non sono in grado di innovare e competere”. Ne consegue, conclude il documento, un impegno maggiore da parte dell'OSAIA per informare il Congresso sulle concrete potenzialità del software open source e sulla piena legalità della licenza GNU GPL all'interno delle attuali norme sul copyright.

D'altronde analoghe accuse a tutto campo contro il fronte del software non proprietario McBride le aveva lanciate fin

⁵ <http://www.osaia.org/modules.php?name=Content&pa=showpage&pid=10>

dallo scorso autunno, come necessario puntello per le contemporanee iniziative legali. In una lettera aperta del 9 settembre 2003⁶, il responsabile di SCO richiamava il mondo dell'open source e del software libero al rispetto “delle regole e dei principi che stanno dietro al concetto di proprietà intellettuale”, esortando i leader e i programmatori di tale mondo a implementare un business model sostenibile, capace di offrire “stabilità e attendibilità alla comunità open source” all'interno del corrente mercato commerciale e professionale. Pur dichiarandosi aperto e disponibile alla collaborazione, McBride aggiungeva tuttavia di dover continuare “a proteggere le proprietà intellettuali e i diritti contrattuali dell'azienda”.

La lettera si concentrava tuttavia su due questioni calde: gli attacchi Denial of Service (DoS) subiti dal sito SCO qualche giorno prima e presunte ammissioni della comunità open source sulla presenza di codice Unix nel kernel Linux. Rispetto a quest'ultimo punto, McBride citava in maniera incompleta un'affermazione di Bruce Perens, influente figura dell'open source, per ribadire il concetto: “Il fatto è che uno sviluppatore Linux stipendiato da Silicon Graphics Inc. ha infranto il copyright di una porzione di codice proveniente da System V, concessa in licenza a SGI sotto rigide condizioni d'uso, includendola in Linux come se si trattasse di codice pulito, sotto la proprietà e il controllo di SGI. Si tratta di una chiara violazione dell'impegno contrattuale, dei vincoli di SGI rispetto al copyright di SCO”. In realtà Perens aveva definito la vicenda, limitata al codice relativo all'allocazione della memoria, “un errore del processo di sviluppo di Linux

⁶ <http://www.sco.com/company/openletter/>

(presso SGI)”. Questa, d’altro canto, si era affrettata a implementare immediatamente delle alternative al codice contestato, nella stessa giornata del primo ottobre 2003 quando SCO dichiarava ufficialmente decaduto il contratto di utilizzo di System V per violazione dei termini della licenza. Ampliando simili specificità del caso SGI, la lettera passava a sottolineare “le grosse falle strutturali nel processo di sviluppo di Linux”, mettendo in dubbio l’applicabilità dell’intero modello ed esigendo anche dal codice open source la garanzia nel rispetto delle leggi sul copyright.

Altro punto cruciale della lettera aperta, le ulteriori accuse agli hacker open source per gli attacchi Denial of Service (DoS) subiti dal sito SCO. Questo era divenuto irraggiungibile per diversi giorni, verso fine agosto, come pure era successo a maggio. In entrambi i casi, McBride sa a chi dare la colpa. “Non esiste alcun dubbio sull’affiliazione dell’aggressore: come riportato da alcune testate (elettroniche), il leader open source Eric Raymond ha sostenuto di essere stato contattato dall’esecutore dell’attacco definendolo ‘uno dei nostri’. A parziale merito di Raymond, ha chiesto a costui di fermarsi. Tuttavia deve ancora rivelarne l’identità in modo che possa esser fatta giustizia”. Insistendo sulla necessità che l’intera comunità si adoperi per contrastare questo tipo di crimini, McBride scrive: “Non possiamo tollerare una situazione in cui le aziende debbano temere di essere i prossimi bersagli di un attacco, nel caso in cui abbiano un business o una posizione legale che indisponga la comunità open source. Finché tali attacchi illegali non verranno messi sotto controllo, le grandi società tenderanno a distanziarsi da chiunque sia associato a questo tipo di comportamenti”.

Ramoscelli d'ulivo e beghe finanziarie

Interessanti alcuni commenti a latere inclusi in un'intervista con il CEO di SCO, apparsa su *ComputerWorld*⁷ un paio di giorni dopo la diffusione della lettera aperta. Quest'ultima veniva definita "un'offerta di pace, un ramoscello d'ulivo" teso alla comunità open source. "Se esiste un modo per risolvere le differenze così da poter andare avanti pacificamente nella risoluzione delle dispute in tempi brevi, noi siamo pienamente d'accordo". Un tono conciliatorio, dunque, anche per superare l'ennesimo impasse causato dai contrattacchi lanciati da IBM e Red Hat in occasione dello SCO Forum, evento per clienti e rivenditori aziendali svoltosi in agosto a Las Vegas. Alla domanda dell'intervistatore su cosa è lecito attendersi in replica, McBride fa chiarezza: "La cosa più semplice che vorremmo vedere è l'esistenza di un business model intorno a Linux capace di offrire e ricompensare quelle aziende, come la nostra, la cui proprietà intellettuale compare all'interno di Linux. In secondo luogo, occorre prevedere un processo per assicurarsi che il codice Linux venga valutato e protetto a livello di proprietà intellettuale".

A completare il quadro di questa panoramica sulla documentazione non-legale targata SCO, vanno citate almeno altre due comunicazioni,⁸ entrambe risalenti al periodo natalizio: una notifica inviata a diverse società della classifica Fortune 1000 che usano Linux e una lettera spedita ai propri licenziatari perché autocertifichino il non-uso di alcun codice illegale. Nel primo caso, a quelle società 'colpevoli' di usare Linux vengono notificate "addizionali violazioni del copyright SCO" conte-

⁷ <http://www.computerworld.com/printthis/2003/0,4814,84819,00.html>

⁸ <http://punto-informatico.it/p.asp?i=46405>

nute all'interno del sistema operativo open source. Facendo seguito a una precedente comunicazione di maggio, il consulente legale Ryan Tibbitts spiega che alcune application binary interface (ABI) tutelate da copyright "risultano copiate integralmente dal codice di UNIX System V e integrate in Linux senza alcuna autorizzazione e attribuzione di copyright... Ogni parte dei file Linux che contenga il codice in oggetto deve pertanto essere rimossa. Tale codice ABI fa parte di un accordo del 1994 con la University of California di Berkeley e la Berkeley Systems Development Inc."

L'altra missiva ha invece come destinatari altrettanti clienti di SCO, regolarmente in possesso di licenza per il sistema operativo Unix System V. Costoro dovranno confermare per iscritto di non aver violato le clausole di confidenzialità relative al codice Unix avuto in licenza da SCO. Dovranno cioè certificare di non utilizzare alcun codice Unix proprietario all'interno di Linux o di altri software basati su Unix. In caso di mancata risposta, entro gennaio 2004, secondo l'accordo di licenza, SCO ha il diritto di "porre termine al contratto e richiedere al licenziatario di interrompere l'utilizzo del software in questione". In pratica, si tratta di quanto era già accaduto a giugno con la revoca della licenza di Unix AIX a IBM. I licenziatari di Unix System V sarebbero oltre 6.000, secondo la stessa SCO, incluse "alcune delle principali aziende farmaceutiche e finanziarie, del settore dei trasporti, dell'energia, dell'automobile, dell'hardware e del software". Un nuovo, pieno sostegno, dunque, alle iniziative legali avviate nel marzo 2003, come conferma Chris Sontag, dirigente dello SCO Group. "Oggi siamo qui per comunicare formalmente ai licenziatari del codice sorgente UNIX e ad alcuni utenti commerciali Linux che, per quanto concerne le pro-

prietà intellettuali di SCO, sono tenuti al rispetto dei termini legali e del Digital Millennium Copyright Act”.

Un ultimo appunto per alcuni risvolti economici più o meno connessi all'affaire legale. I dati relativi all'esercizio finanziario dello scorso anno per il gruppo SCO riportano guadagni netti per 5,3 milioni di dollari, rispetto a una perdita di quasi 25 milioni di dollari registratasi nel 2002. Il fatturato 2003 è cresciuto del 23 per cento, pari a 79,3 milioni di dollari (64,2 milioni nel 2002). Senza la spesa di 9 milioni di dollari per i compensi allo studio legale guidato da David Boies, SCO avrebbe toccato un utile netto di esercizio di 14,3 milioni di dollari. Si tratta della prima volta che SCO, o la precedente Caldera Systems, riporta un utile sull'attività annuale. Esulta Darl McBride: “L'esercizio 2003 è stato un anno di eccezionale successo per SCO. Abbiamo registrato risultati finanziari da record, compreso il primo anno di piena profittabilità... Questo dà a SCO le risorse e la flessibilità necessarie per continuare a proteggere la propria proprietà intellettuale su Unix ed espandere il relativo business”.

Sempre a livello finanziario, va infine aggiunto un dato comunque preoccupante: sulla base di alcune memo interne, diffuse recentemente da Eric Raymond⁹ e confermate come autentiche dagli stessi dirigenti SCO, Microsoft sarebbe direttamente implicata in sovvenzioni fino a 86 milioni di dollari ricevuti da Caldera/SCO lo scorso ottobre tramite la società d'investimenti californiana Baystar Capital.

Insomma: tutto a posto e niente in ordine. Parola di SCO e Darl McBride.

⁹ <http://www.opensource.org/halloween/halloween10.html>

Le reazioni della comunità del software libero e open source

“La GNU GPL (GNU General Public License) viola la Costituzione statunitense ed elude anche il diritto d’autore, le leggi sull’Antitrust e il controllo sulle esportazioni”¹. Se invece di una causa giudiziaria, il caso SCO fosse uno dei conflitti armati che infuocano il pianeta, l’affermazione, contenuta nella risposta del 24 ottobre 2003 a IBM, sarebbe un colpo di mortaio sparato a bruciapelo, una granata catapultata su un nemico individuato non solo in una multinazionale che, vestita di blu, insidierebbe clienti e strumenti di un concorrente. Il nemico di SCO è (anche e, forse, soprattutto) un sistema filosofico, un’idea, un modello rappresentato, attraverso il sistema operativo finito sotto il tiro incrociato delle trincee costruite dopo il 6 marzo 2003, dalla Free Software Foundation. Una condotta che ricorda il ritrito “colpirne uno per educarne cento”: fare fuoco su Linux e abbatterlo significherebbe, probabilmente, trascinare al collasso l’intera comunità del software libero e open source.

Ma il caso SCO non è una guerra. Su questa vicenda non scorre sangue, scorre denaro. In palio non c’è il dominio di una nazione, ma quote di mercato. Il lavoro sporco non lo fa l’intelligence, ma studi legali di grido, in un gioco di strategia in

¹<http://www.groklaw.net/pdf/AnswerAmendCC.10-24-03.pdf>

cui mancano pedine, le regole sono torbide e gli obiettivi collaterali espliciti. Roba da avvocati, insomma, al soldo di amministratori delegati che se la sono giurata, consci del fatto che, laddove l'imprenditorialità non paga più e la borsa tira ma solo verso il basso, i profitti possono essere generati dalle politiche di "litigation company".

Gli "obiettivi collaterali" – programmatori, attivisti, sostenitori della libertà informatica – non sono stati però ad attendere la sentenza della corte dello Utah. Evitando di impastarsi nella mischia, hanno sfoderato la loro arma migliore, la comunità, che ha lavorato e lavora per raccogliere e organizzare materiale (si veda a titolo di esempio il caso di *Groklaw*² e del costante servizio informativo che sta portando avanti), per verificare affermazioni e per analizzare documenti legali. Un lavoro certosino e scrupoloso, sottoposto al confronto pubblico, di cui si sono incentivate la diffusione e la rielaborazione³. In pieno stile GPL. Non sono mancate poi le voci *ufficiali*, quelle dei leader storici della comunità, che hanno offerto interpretazioni e commenti, controbattendo alle insinuazioni che, con il progredire della vicenda, hanno tentato di minare le basi di vent'anni di attività.

“Una campagna di diffamazione non potrà distruggerci”

Richard Stallman, noto anche con la firma RMS, ha bisogno di ben poche parole di presentazione. Talmente poche che, quando invia un articolo a una serie di testate online, diretto-

²<http://www.groklaw.net/>

³Oltre a *Groklaw*, un altro esempio del fitto lavoro di informazione della comunità è rappresentato dalla sezione SCOVsIBM di TwikIWeThey, definito come "an experiment in collective intelligence. Stupidity. Whatever". Attraverso un sistema di web publishing aperto a chi vuole contribuire e con un limitato intervento di moderazione, si è lasciato che fossero i lettori ad aggiornare il materiale che riguardava il caso.

ri e redattori si affrettano a pubblicarlo. È il caso di “SCO smear campaign can't defeat GNU community”⁴ (“La campagna di diffamazione di SCO non può scalfire la comunità GNU”), datato 23 giugno 2003, una delle prime reazioni ufficiali di fronte alle accuse secondo cui Linux sarebbe un clone di Unix. Parole fitte ma semplici per sostenere che “SCO sta usando la comune confusione tra Linux ed il sistema operativo GNU/Linux per ingigantire il timore che cerca di seminare”. Perché, come spiega il fondatore della Free Software Foundation, “Unix è un sistema operativo completo, mentre Linux ne è solamente una parte [...]. Il kernel è la porzione del sistema che alloca le risorse della macchina agli altri programmi che ci girano sopra”.

E qui la prima puntualizzazione tecnica. Una puntualizzazione che fa sorridere geek e hacker. Cioè quei tecnofili curiosi e preparati, per semplificare le definizioni usate dal *Jargon File*⁵, che vorrebbero sbagliare quando pensano di dover spiegare le basi alle holding dell'informatica, concetti che queste dovrebbero già aver interiorizzato. Proseguendo nella lettura del documento, Stallman ripercorre la storia del software libero, passa-

⁴La versione originale dell'articolo di Richard Stallman è apparso per la prima volta su ZDNet all'indirizzo

<http://techupdate.zdnet.com/techupdate/stories/main/0,14179,2914132,00.html>.

Successivamente è stato pubblicato anche sul sito della Free Software Foundation con il titolo SCO, GNU and Linux e la traduzione in italiano del testo, curata dal team The Gnu Project, è stata pubblicata all'indirizzo <http://www.fsf.org/philosophy/sco/sco-gnu-linux.html>.

⁵Il Jargon File è un testo di riferimento negli ambienti informatici per quanto riguarda le definizioni tecniche e trova il suo corrispettivo cartaceo nel The New Hacker's Dictionary. Le risorse da esso comprese sono reperibili all'indirizzo

<http://www.catb.org/~esr/jargon/>. Il testo, si legge sulla home page del progetto curato da Eric Raymond, ha avuto “il dubbio onore di essere citato nella causa SCO contro IBM” (<http://www.catb.org/~esr/jargon/html/F/FUD.html>) il 23 ottobre 2003.

ta attraverso la nascita del Progetto GNU nel 1984, l'avvio dello sviluppo di Linux nel 1991 e le implicazioni collegate al suo passaggio da una licenza per scopi non commerciali alla GNU GPL. Riferendosi poi al “termine alla moda ma insensato” di “proprietà intellettuale” e al pregiudizio che genera trattando idee, lavori e nomi come se potessero essere posseduti alla stregua di oggetti materiali, RMS traccia un'immagine chiara del suo pensiero sulla vicenda.

“In una reale causa legale, una tale ambiguità farebbe fallire l'istanza o addirittura la impedirebbe del tutto. Se lo scopo di SCO è comunque quello di scuotere l'albero per vedere se ne cade qualcosa o semplicemente quello di diffondere paure, potrebbe considerare ambiguità e mistero come fattori vantaggiosi [...]. SCO non può usare i propri copyright o i propri contratti come strumenti per sopprimere i legittimi contributi di migliaia di altri soggetti [...]”.

Ad intervenire più di frequente e con una serie⁶ di articoli è poi Eben Moglen, docente di legge alla Columbia University Law School e consulente legale della Free Software Foundation. Lo fa la prima volta il 25 giugno 2003 con il documento “FSF Statement on SCO v. IBM”⁷ (“Dichiarazione di FSF su SCO contro IBM”). E aggiunge una serie di elementi importanti, oltre a quanto già spiegato da Stallman, partendo dal tentativo di SCO di voler colpire tutto il software libero tramite Linux:

“Il sistema operativo contiene molte altre componenti [oltre al kernel], la maggior parte dei quali sono prodotti del Progetto

⁶*L'elenco completo degli articoli pubblicati dalla Free Software Foundation sul caso SCO si trova all'indirizzo <http://www.fsf.org/philosophy/sco/sco.html>.*

⁷*<http://www.fsf.org/philosophy/sco/sco-v-ibm.html>*

GNU della Free Software Foundation [...] che includono il compilatore GCC, il debugger GDB, la libreria Glibc, la shell bash, per limitarsi alle parti essenziali. La combinazione di GNU e del kernel Linux produce il sistema GNU/Linux, ampiamente usato su differenti architetture hardware che, *prese nel loro insieme*, riproducono le funzioni di un sistema operativo Unix. L'uso confuso che SCO fa dei nomi pone le basi per le sue oscure pretese: SCO ha dichiarato che i segreti commerciali del creatore di Unix, AT&T – di cui SCO, per transazioni intermedie, è il successore interessato – sono stati incorporati da IBM nel kernel Linux o in una parte del software GNU? Se l'opzione è la prima, non trovano giustificazione le dichiarazioni che raccomandano cautela nell'uso di software libero o di programmi GNU. Se, invece, SCO sostiene che GNU contenga qualsivoglia segreto commerciale o materiale di UNIX sotto copyright, allora è quanto meno in malafede”.

Moglen prosegue elencando altre ragioni per cui le asserzioni di SCO sarebbero prive di fondamento. Tra queste il fatto che SCO ha distribuito per anni copie di Linux come parte del sistema libero GNU/Linux.

“Questi sistemi sono stati distribuiti in piena osservanza della GPL, includendo quindi il codice sorgente completo. Come SCO ha pubblicizzato in continuazione [...], quel materiale include anche segreti commerciali. Semplicemente non c'è alcuna base legale che permetta a SCO di rivendicare segreti su materiale che è ampiamente e commercialmente pubblicato con una licenza che permette in termini non restrittivi la copia e la distribuzione. Un fatto, questo, che pone una irrevocabile barriera alla rivendicazione secondo cui “Linux” violi il copy-

right di SCO attraverso il codice sorgente di Unix. Il copyright, come enfatizzato più volte dalla Corte Suprema degli Stati Uniti, copre *espressioni*, non *idee*".

Intanto SCO, dopo aver "condonato" agli utenti privati l'installazione di Linux, scrive alle aziende: se hanno installato il sistema operativo libero, possono "regolarizzare" la loro posizione pagando una licenza di appena 699 dollari. Puntuale arriva il commento di Moglen. È il primo agosto e in "Questioning SCO: A Hard Look at Nebulous Claims"⁸ ("Gli interrogativi di SCO: un rigido sguardo a pretese nebuloze"), esordisce affermando che "un'asserzione apparentemente seria è stata formulata in modo essenzialmente non serio: con un comunicato stampa non accompagnato da alcuna prova che possa permettere la formulazione in un giudizio fondato". Come sottolinea ancora il docente della Columbia University:

"L'iter usuale in cause di violazione del diritto d'autore prevede l'indicazione del distributore o dei distributori del lavoro frutto della presunta violazione e l'esibizione del lavoro violato. SCO non l'ha fatto. Per esempio, in un comunicato stampa del 21 luglio ha asserito che il codice contenuto in versioni recenti di Linux per il symmetric multi-processing viola il loro diritto d'autore [...]. Il supporto SMP è per la maggior parte opera di frequenti contributi di dipendenti RedHat e Intel. SCO non ha presentato i sorgenti che questi programmatori avrebbero copiato e non ha fatto causa a nessuno. Invece ha chie-

⁸<http://www.fsf.org/philosophy/sco/questioning-sco.html>. Il documento è stato ripreso dall'OSDL (Open Source Development Labs) per farne anche la propria posizione di fronte all'attacco di SCO alla comunità (http://www.osdl.org/docs/osdl_eben_moglen_position_paper.pdf).

sto agli utenti di acquistare una licenza [...]. Questi utenti [...] hanno il diritto di porre una serie di domande. In primo luogo, dove stanno le prove? Che cosa è stato copiato dal lavoro di SCO? In seconda istanza, perché avrei bisogno di una licenza per usare l'opera senza preoccuparmi di chi sia il detentore dei diritti su ciascuna parte? Terzo, non avete distribuito voi stessi quest'opera con una licenza che permette a tutti, anche a me, di copiarla, modificarla e distribuirla liberamente?”.

La speranza con cui Eben Moglen chiude l'articolo (“sentire presto qualche risposta”) va delusa e il 18 agosto, nell'articolo “SCO Scuttles Sense, Claiming GPL Invalidity”⁹ (“SCO smarrisce il senso, rivendicando l'invalidità della GPL”), ne commenta il tentativo di “rafforzare i propri diritti di proprietà intellettuale”, come reazione alla progressiva perdita di fondamento della causa. La settimana precedente, infatti, Mark Heise, consigliere dell'azienda querelante, aveva ribadito al *Wall Street Journal* la sfida alla “legalità” della GPL affermando che, sì, la licenza consente un numero infinito di copie, ma che la legge sul diritto d'autore concederebbe invece una singola copia. “Argomento frivolo”, liquida l'affermazione Moglen, che passa a commentare la sezione 117 dello US Copyright Act intitolato “Limitation on exclusive rights: computer programs” (“Limitazioni sui diritti di esclusività: programmi informatici”).

“La sezione 117 sostiene che, sebbene la legge proibisca generalmente la creazione di copie di un lavoro sotto copyright, nel caso di programmi informatici è possibile copiare e modificare l'opera. Pretendere che questa disposizione fissi un limite su ciò che i detentori dei diritti possono permettere nel licenziare

⁹<http://www.fsf.org/philosophy/sco/sco-preemption.html>

tali diritti è del tutto falso. Non trova riscontro nel linguaggio legale, nella storia legislativa, nella giurisprudenza o nella politica costituzionale. Una motivazione del genere, presentata a una corte, verrebbe sicuramente giudicata inconsistente”.

I toni di Moglen si fanno via via meno morbidi con il progredire della vicenda. È il 24 novembre quando esce un suo nuovo articolo, “SCO: Without Fear and Without Research”¹⁰ (“SCO: senza paura e senza ricerca”), che esordisce con “la definizione tradizionale di azzecagarbugli: un avvocato che, quando la legge è contro di lui, se la prende con le prove; quando le prove sono contro di lui, se la prende con la legge; e quando sia la legge che le prove sono contro di lui, prende a pugni il tavolo”. L’invettiva è contro la pratica di SCO di aumentare il proprio valore di mercato a spese di sviluppatori, distributori e utenti di software libero, “attraverso bizzarre teorie legali. [Le loro] rivendicazioni prive di riscontri effettivi confermano che quella vecchia definizione non ha perso efficacia”. A scatenare la reazione dell’avvocato è stata l’uscita pubblica, il 18 agosto precedente, di Darl McBride, che ha presentato una relazione che avrebbe dovuto contenere due nuovi elementi ad avallo della presunta violazione. Il primo è il firewall chiamato Berkeley Packet Filter (BPF): una sua implementazione è contenuta nel kernel di Linux, ma è di Jay Schulist il lavoro originale ed è stato inserito successivamente in System V di SCO prendendo codice dalla versione di BSD (Berkeley Software Distribution). Dunque, sia Sys V che Linux hanno un genitore in comune per l’esecuzione di quella funzione e il codice prodotto da quel genitore non è di proprietà di SCO. La seconda “prova” riguarda

¹⁰ <http://www.fsf.org/philosophy/sco/sco-without-fear.html>

una dozzina di linee di codice per l'allocazione della memoria, identiche nei due sistemi operativi. Peccato che il codice in C mostrato da McBride risalga al 1973 e discenda direttamente dalla versione pubblicata nel 1968 da Donald Knuth in "The art of computer Programming" ("L'arte di programmare"). Risulta poi irrilevante per il CEO che ad accaparrarsi la proprietà di quella stessa porzione di codice ci avesse inutilmente già provato AT&T nella causa contro BSD e che la sentenza avesse stabilito quanto segue: risalendo a prima del 1976 e dunque non potendo essere compreso sotto la legge sul diritto d'autore americana, quel sorgente andava considerato di pubblico dominio. Tirando le conclusioni, Moglen sostiene che:

"La risposta di SCO a IBM è centrata su un attacco assurdo alla GPL, ai suoi utenti e ai suoi autori, oltre che alla Free Software Foundation. La GPL, secondo SCO, viola non solo le leggi federali, ma anche la Costituzione statunitense. Come possa un soggetto privato violare la Costituzione se dà il permesso di copiare, modificare e ridistribuire il proprio lavoro non è dato saperlo. Le teorie legali non sono segrete; se gli avvocati di SCO hanno qualche asso a suffragio di questa tesi romanzesca, che li mostrino. Non esiste un solo caso nella storia del copyright americano che si fondi sulla ridicola concezione di una licenza incostituzionale. Dopo aver fallito con i fatti e con la legge, aver ricavato solo risate dai pugni sul tavolo, anche il proverbiale azzecagarbugli non sa più che fare".

Il ritorno di Halloween

Esiste un filone cinematografico che, a suon di sequel, ha trasformato pellicole da incubo in fenomeni cult. È il caso di *Venerdì 13*, *Nightmare* o *La casa*. È anche il caso di *Halloween*,

la storia di Michael Myers, che, dopo aver accoltellato a sei anni la sorella e il fidanzato di lei e aver trascorso il successivo ventennio in manicomio, scappa e scatena un'ondata di terrore e sangue fino a diventare, con il progredire della saga, una maligna presenza altrettanto implacabile e, ovviamente, immortale.

È la notte delle streghe ad aver ispirato il regista, John Carpenter. Ed è sempre la stessa ricorrenza che ha suggerito ad Eric Raymond, presidente della Open Source Initiative, il titolo per una serie di saggi basati su documenti originariamente usciti dagli uffici della Microsoft alla fine dell'ottobre 1998. In questo caso, il terrore non è così accecante come quello di adolescenti della provincia USA fatti a pezzi da un serial killer e l'autore dei crimini non è uno psicopatico sfuggito al controllo della psichiatria moderna. I documenti di Halloween¹¹, almeno per quanto riguarda i primi tre, manifestano un altro tipo di inquietudine – quella del colosso di Redmond verso “una nuova metodologia di sviluppo” – e la minaccia derivante da un sistema, quello open source, che scardina le basi commerciali dell'industria del software proprietario.

Nati dunque come raccolta di documenti anti-Microsoft, gli Halloween si sono evoluti verso testimonianze della concreta efficienza di Linux e del software open source in generale. Se il documento numero otto è datato 25 novembre 2002 e torna al vecchio leitmotiv dei memorandum fuoriusciti dalla rete di Bill Gates, il nove va online il 28 giugno 2003 e, cambiando obiettivo aziendale e titolo, “Halloween IX: It Ain't Necessarily

¹¹*L'elenco completo dei documenti di Halloween è disponibile all'indirizzo <http://www.opensource.org/halloween/>.*

SCO”¹² (“Halloween IX: non è necessariamente SCO”), fa capire subito verso chi sia puntato. Gli autori iniziali sono di nuovo Raymond e lo sviluppatore Rob Landley, ma il successo del documento è tale che all’ultima release (versione 1.3, datata 28 agosto 2003) il numero di persone che vi ha man mano contribuito sfiora la trentina. Fin dall’introduzione, il testo non lascia adito a dubbi sull’impostazione generale: la vicenda è intessuta di “menzogne e deliberate distorsioni”. E prosegue riportando tutti i passaggi salienti della querela: la causa per la violazione dell’accordo sul software, quella per la sottoliscenza, il contratto di Sequent, il comportamento anti-competitivo e le interferenze contrattuali. Per finire ai fiaschi dell’Itanium e di Monterey, che, secondo SCO, sono stati una manovra di IBM per farne naufragare le velleità verso un nuovo Unix proprietario con cui rimpiazzare i datati OpenServer e UnixWare.

Più recente il documento “Halloween X: Follow The Money”¹³, (“Halloween X: la pista del denaro”) pubblicato il 3 marzo 2004. La sua forma torna alla classica mail sfuggita al controllo di mittenti e destinatari, la cui autenticità viene confermata a ventiquattr’ore di distanza dalla stessa SCO. Il mittente è Mike Anderer, CEO di S-2 Strategic Consulting, azienda di Salt Lake City che, con BayStar Capital, ha gettato a SCO un salvagente finanziario di 50 milioni di dollari – questa la cifra ufficiale, ma si vedrà come quelle reali siano differenti. Il destinatario del messaggio di posta elettronica risulta invece essere Chris Sontag, vice presidente e general manager di SCOSource, divisione responsabile dello sviluppo e del licenziamento dei

¹²<http://www.opensource.org/halloween/halloween9.php>

¹³<http://www.opensource.org/halloween/halloween10.html>

prodotti di proprietà dell'azienda dello Utah. In copia carbone Bob Bench, dirigente economico di SCO Group. E, attraverso un calcolo fittizio di percentuali su accordi, brevetti e licenze, viene fuori quanto ghiotta sia l'attività di SCO per qualche "grande nome". Anderer parla di un finanziamento Microsoft di 86 milioni di dollari, molti di più di quanto SCO stessa avesse ammesso (68,5), di un'ulteriore iniezione di 16-50 milioni attraverso "accordi incrementali"¹⁴ e di operazioni che avrebbero avuto lo scopo di stornare denaro dalla divisione sviluppo a quella marketing. Il linguaggio della mail è spesso oscuro, gli acronimi che indicano alcune società possono essere solo desunti¹⁵ e le differenze tra gli stanziamenti ufficiali e quelli riportati nel messaggio lasciano trapelare una complessa operazione finanziaria, condotta attraverso venture capitalist usati come paravento a copertura delle vere fonti dei flussi di denaro. Ma le manovre a cui si allude permettono di intuire che la SEC (Securities and Exchange Commission) stia indagando sui movimenti partiti da Microsoft alla volta di Salt Lake City¹⁶. Un'ulteriore analisi della situazione, condotta in termini storici e tecnici ma non economici, realizzata per un altro documento disponibile sul sito della Open Source Initiative (OSI): si tratta della "Position Paper on the SCO-vs.-IBM Complaint"¹⁷, ("Posizione sulle rivendicazioni di SCO contro

¹⁴Il riferimento è a Red Hat, Acrylis, società coinvolta nell'acquisizione del metodo di acquisto WhatIfLinux, e IPX, società creata per supportare SCO nella causa contro Novell per questioni di brevetti.

¹⁵Quando si cita una non meglio definita MCS, si intende probabilmente Microsoft Consulting Services.

¹⁶A questo proposito, una parziale conferma arriva dall'articolo comparso l'8 marzo 2004 su Newsforge, Analysis: Microsoft, SCO have a lot more explaining to do (<http://trends.newsforge.com/trends/04/03/08/0457259.shtml>).

¹⁷<http://www.opensource.org/sco-vs-ibm.html>

IBM”) ancora una volta firmata da Eric Raymond. Anche in questo caso è un’analisi in fieri: la prima versione è stata pubblicata il 10 marzo 2003 e la più recente, al momento in cui si scrive, risale al 22 agosto 2003 con la release 2.4. E pone una domanda interessante: chi è il vero proprietario di Unix, il sistema tanto conteso fin dalla causa che aveva visto contrapposti AT&T e l’Università della California? Sicuramente non SCO né la precedente Caldera, in base alla ricostruzione di Raymond, il quale raccomanda alla corte, nel paragrafo successivo, di tenere conto di una serie di elementi:

“La segretezza commerciale nel software è incompatibile con le pratiche commerciali della vendita di licenze su codici sorgenti di software destinato a istituzioni educative o per usi formativi. [Inoltre] a ogni rivenditore che abbia beneficiato di prodotti open source dovrebbe essere vietata qualsiasi pretesa sui diritti di proprietà per impedire ad altri partecipanti di godere degli stessi benefici [...]. La GNU General Public License con cui viene rilasciato Linux è una licenza valida e impone obblighi precisi a chi ne accetta i termini”.

La vicenda è riuscita altresì a compiere una specie di miracolo: quello di riunire Raymond all’amico-nemico Bruce Perens, uno degli autori della Open Source Definition ed attualmente direttore dell’organizzazione non-profit Software in the Public Interest. I due firmano infatti congiuntamente la risposta alla “Open Letter to the Open Source Community”¹⁸ (“Lettera aperta alla comunità open source”) diffusa da Darl McBride il 9 settembre 2003. Qui il CEO di SCO accusa esplicitamente

¹⁸ Il testo della risposta di Perens e Raymond si trova all’indirizzo

<http://www.newsforge.com/software/03/09/09/2355214.shtml?tid=11> mentre la lettera aperta di McBride a <http://www.newsforge.com/business/03/09/09/1327245.shtml?tid=17>.

la comunità di essere il mandante dell'ennesimo Denial of Service (DoS) lanciato contro il sito della sua azienda e torna a rimarcare la proprietà di parte del codice di Linux. In meno di 24 ore, i due padri dell'OSI fanno notare che proprio dal movimento open source arriva la richiesta di smetterla con i DoS. Ma soprattutto, rispediscono al mittente qualsiasi offerta di trattativa.

“Linux è lavoro *nostro* ed è legalmente di nostra proprietà, il distillato di dodici anni di duro impegno, idealismo, creatività, lacrime, gioia e sudore di centinaia di migliaia di hacker che cooperano in tutto il mondo [...]. Se vuoi rendere rispettabile la causa legale, *tira fuori il codice*. Scopri le coincidenze. Specifica file per file e linea per linea quale sarebbero i sorgenti violati”.

Mentre Raymond lavorava ai nuovi documenti di Halloween, per conto suo nemmeno Perens tralasciava commenti e analisi. Lo testimonia, ad esempio, la ricerca “Analysis of SCO's Las Vegas Slide Show”¹⁹ (“Analisi sulle diapositive di SCO presentate a Las Vegas”) in cui l'autore commenta, una diapositiva dopo l'altra, la già citata presentazione di McBride spiegando nel dettaglio l'origine e l'evoluzione di elementi come il Berkeley Packet Filter o il Journaling File System, nato originariamente per OS/2 e poi portato su Linux. Negli articoli “SCO's Big Lie”²⁰ (“La grande menzogna di SCO”) ripercorre poi l'estensione a Novell della querela, sottolineando come quest'ultima non sia coinvolta nelle contestazioni di SCO ed enfatizzandone il rapporto di reciproco supporto con la comunità open source/software libero. Infine, in “SCO's Tapestry of

¹⁹ <http://www.perens.com/SCO/SCOSlideShow.html>

²⁰ <http://www.perens.com/SCO/BigLie.html>

Lies”²¹ (“L’arazzo delle bugie di SCO”), arriva all’analisi del coinvolgimento di AutoZone, società di Memphis che produce accessori e software diagnostici automobilistici, ‘rea’ di essere diventata cliente di RedHat e IBM passando a Linux, e di Daimler Chrysler, ennesimo violatore di presunti accordi contrattuali. E torna l’accenno finanziario.

“La campagna di SCO contro Linux dura ormai da un anno ed ha consentito il decollo delle sue azioni da cinquanta centesimi a più di venti dollari [...]. Quando un’azienda formula asserzioni prive di fondamento per un mese o due, si può credere in un errore o in un’illusione. Quando le distorsioni vanno avanti per un anno intero, diventa difficile spiegare un comportamento se non ricorrendo alla frode deliberata”.

“Siete come la televisione spazzatura”

Chi l’avrebbe mai detto che da oscuro studente di informatica Linus Torvalds si sarebbe ritrovato ad essere il programmatore più famoso del mondo e, successivamente, l’involontaria miccia di un caso giudiziario molto più vasto ed aggressivo del precedente AT&T contro BSD? Il padre di Linux non è sembrato però mai particolarmente preoccupato per le sorti del sistema operativo da lui ideato. Anzi. Nella lettera aperta del 10 settembre 2003 a McBride, “Please grow up”²² (“Per favore, crescete”), scrive:

“Siamo felici che concordi sulla necessità di far sapere ai clienti che l’open source è legale e stabile [...], ma abbiamo sfortunatamente declinato il consiglio su un modello di business arrivato da un’azienda che sembra aver dissipato tutti i suoi quat-

²¹<http://east.perens.com/SCO/March2004.html>

²²<http://www.newsforge.com/software/03/09/10/2321224.shtml?tid=11>

trini [...] e ora sembra giocare con il sistema legale americano come se fosse una lotteria. Noi continuiamo a credere nella tecnologia come strada per guidare l'interesse e la richiesta dei clienti. Inoltre, le tue proposte di negoziare ci appaiono confuse dato che non vediamo nulla su cui negoziare. SCO non ha dimostrato nessuna violazione da parte dell'open source, ma attendiamo con il fiato sospeso il momento in cui ci informerai di ciò che vai blaterando”.

Vetriolo come se piovesse, dunque, come d'altronde accade fin dall'inizio della disputa. Il 30 maggio, infatti, in un'intervista condotta via e-mail, Torvalds paragona la battaglia tra SCO, IBM e Novell alla “televisione spazzatura”²³. E già dal titolo, “Open-source experts deride SCO's 'evidence'”²⁴ (“Gli esperti dell'open source deridono le prove di SCO”), il testo del 20 agosto non dimostra alcun cambiamento di tono, ribadendo “che l'abbaglio [di SCO] non è comunque una sorpresa”. In una precedente intervista, condotta l'11 luglio e pubblicata con il titolo “SCO ne me fait pas peur”²⁵ (“SCO non mi fa paura”), affronta la questione in modo più approfondito e più pacato, anche se forse non meno infastidito. Parlando dei sistemi di verifica e di validazione del codice e delle conseguenze della disputa giudiziaria sul lavoro quotidiano, Torvalds insiste: “Disponiamo già da tempo di sistemi di verifica, ma consentono soprattutto di commentare le nuove funzioni. Non sono in alcun modo legate a questioni di proprietà intellettuale”. Sulle quali invece aggiunge:

²³ Si veda l'articolo *Analysts to SCO: No thanks to code review offer* (<http://www.computerworld.com/softwaretopics/os/linux/story/0,10801,81695,00.html>).

²⁴ <http://news.zdnet.co.uk/software/linuxunix/0,39020390,39115818,00.htm>

²⁵ <http://www.zdnet.fr/actualites/technologie/0,39020809,2137439,00.htm>

“Francamente inizio ad averne abbastanza di tutte queste storie sulla ‘proprietà intellettuale proprietaria’. Quante volte occorrerà ripetere che l’affare SCO non ha niente a che vedere con tutto ciò, che SCO non fa che dire non si sa bene che sulla sua tossica strategia riguardo la ‘proprietà’. È FUD [*Fear, Uncertainty and Doubt*] per instillare paura e incertezza”.

La compattezza della comunità

La causa SCO un effetto sicuro l’ha avuto: ricompattare la comunità anche laddove essa si divide in correnti più o meno opposte. La Free Software Foundation con la Open Source Iniziative, Eric Raymond e Bruce Perens, Linus Torvalds e Richard Stallman. A suffragio di un elemento di base: a prescindere dal fatto che ci si focalizzi principalmente sugli ideali o sul mercato, che si ci ritenga meno ingegneri e più divulgatori o che si preferisca l’illuminazione di un LCD ai riflettori della stampa, lo scopo di fondo rimane comune e identico. Il software libero deve essere libero e disponibile, non ci sono interessi commerciali o investimenti da capogiro che tengano. E si conferma un altro elemento importante: la pronta reattività della comunità che, seppur non proprio spaventata, ci tiene a puntualizzare una situazione che potrebbe apparire oscurata da ombre.

Lo dimostra la pioggia di documentazione che si può leggere sul web. Citarla tutta significherebbe riempire almeno un libro intero solo di indirizzi Internet. Quelli che seguono vogliono essere un semplice spunto che si va ad aggiungere ai documenti e agli articoli segnalati e/o utilizzati nell’arco di questo volume. L’ipertesto è un’invenzione splendida e la comunità del software libero la usa fino in fondo perché l’informazione – che deve

essere non meno che libera per natura – possa circolare nella maniera più capillare possibile.

SCO lawsuit: more conspiracy theories than the JFK assassination:
<http://wagblog.internetweek.com/archives/001184.html>

Group: Backgrounder on the UNIX System and SCO / IBM legal action: <http://www.opengroup.org/comm/press/unix-backgrounder.htm>; <http://www.linux.com/article.pl?sid=03/05/22/2236230>

Text of SCO's complaint: <http://lists.alt.org/pipermail/fsl-discuss/2003-March/000763.html>

Team questions SCO's stance on GPL: <http://www.smh.com.au/articles/2003/08/20/1061261198262.html>

Vs. SCO: The Decision Matrix:
http://www.cybersource.com.au/users/conz/linux_vs_sco_matrix.html

Associazione Software Libero, *Chiarimenti relativi alla nota diatriba che oppone le compagnie informatiche SCO a IBM e Novell:* <http://www.softwarelibero.it/news/20040222.shtml>

Associazione Software Libero, *La causa tra SCO e IBM:* http://www.softwarelibero.it/progetti/scosaga/sco_saga-1.3.html

Comunicato FSF Europe - capitolo italiano, *La Free Software Foundation difende la GNU GPL dagli attacchi di SCO:* <http://mail.fsf-europe.org/pipermail/press-release-it/2004q1/000020.html>

SCO contro IBM: l'inutile guerra contro il codice aperto: prima parte, <http://www.interlex.it/copyright/scovsibm1.htm>; seconda parte, <http://www.interlex.it/copyright/scovsibm2.htm>

Cronologia

6 marzo 2003: SCO cita in giudizio IBM per violazione delle norme contrattuali e l'utilizzo del suo codice proprietario nel kernel di Linux¹. Partendo dal presupposto che Linux sia un derivato di Unix, SCO attacca l'azienda concorrente, "colpevole", secondo l'accusa, di aver tirato la volata – violando contratto e segreto commerciale – al sistema operativo di Linus Torvalds rendendolo adatto per il mercato enterprise. Lo avrebbe fatto implementando specifiche funzionalità, come il supporto per multiprocessore e i file system journaled, attraverso il codice di AIX che IBM non avrebbe potuto utilizzare. Come riferito da varie fonti, tra cui *Linux.org*², le accuse originali sembrano essere "appropriazione indebita di segreti commerciali, concorrenza sleale, violazione di contratto e interferenza illecita con gli affari di SCO". La richiesta iniziale è di un miliardo di dollari. Quali sarebbero nel dettaglio le presunte "colpe" di IBM? Il supporto per JFS, un file system journaled per Linux, che tra l'altro era stato sviluppato originariamente per OS/2³; il sotto-sistema NUMA (Non-Uniform Memory Access), che serve per gestire l'accesso preferenziale alla memoria locale di ogni CPU nei sistemi multiprocessore; l'algoritmo RCU (Read-Copy-Update), che riduce i lock quando più CPU devono leggere (ed eventualmente modificare) gli stessi dati⁴; il sotto-sistema SMP (Symmetric Multi-Processors)⁵.

¹La natura dell'azione di SCO, <http://www.sco.com/scosource/complaint3.06.03.html>

²SCO Controversy Timeline, <http://www.linux.org/news/sco/timeline.html>

³A mini-FAQ for JFS, <http://www-124.ibm.com/developerworks/loss/jfs/project/publfaq.txt>

⁴What is Read-Copy Update?, <http://lse.sourceforge.net/locking/rcu/HOWTO/intro.html#WHATIS>

⁵SCO Owns Your Computer, http://www.byte.com/documents/s=8276/byt1055784622054/0616_marshall.html

10 marzo: prima reazione dell'Open Source Initiative con un *amicus curiae brief* in favore di IBM scritto da Eric S. Raymond e Rob Landley⁶. È sempre di questa data la prima reazione pubblica di Linus Torvalds, in un'intervista a *MozillaQuest*⁷. Il quale afferma sostanzialmente che il caso sembra interessare soltanto i contratti tra SCO e IBM, e che Linux ne viene toccato soltanto in maniera marginale. Secondo Torvalds, SCO sta gonfiando l'importanza di IBM nello sviluppo del sistema open source.

25 marzo: David Boies, Stephen Zack and Mark Heise, gli avvocati di SCO, preparano il dispositivo con i requisiti per la comparizione. Il giorno successivo il caso viene assegnato al giudice Dale Kimball e, subito dopo, arriva un memoriale per la ricusazione firmato dal giudice Paul Cassell⁸. Alcune curiosità: David Boies è stato, tra le altre cose, tra gli avvocati che si sono occupati di rappresentare Al Gore nell'affaire elettorale in Florida nelle presidenziali del 2000⁹ e ha rappresentato il dipartimento di giustizia nel primo caso Antitrust contro Microsoft, conclusosi con il verdetto di smembramento¹⁰. Sempre lui ha difeso Napster nella causa intentatagli dalle major discografiche¹¹. *Time Magazine* lo ha nominato "avvocato dell'anno 2000"¹². Sempre il 25 marzo,

⁶OSI Position Paper on the SCO-vs.-IBM Complaint, <http://www.opensource.org/sco-vs-ibm.html>

⁷SCO-Caldera v IBM: Linus Torvalds Comments on SCO-Caldera's Linux-Related Allegations, http://www.mozillaquest.com/Linux03/SCO-Source-05_Story01.html

⁸IBM Timeline,

<http://www.groklaw.net/staticpages/index.php?page=20031016162215566>

⁹Microsoft-Buster Joins Gore's Legal Team In Florida,

<http://www.techweb.com/wire/story/TWB20001114S0018>

¹⁰DOJ taps outsider for MS case, <http://news.com.com/2100-1023-206522.html?legacy=cnet>

¹¹Music fans hope attorney David Boies can keep downloads free, <http://news.com.com/2009-1023-252774.html>

¹²David Boies. Lawyer of the Year, <http://www.time.com/time/poy2000/boies.html>

IBM ha chiesto di spostare il caso dal tribunale dello Stato dello Utah al tribunale federale¹³.

30 aprile: IBM replica alle accuse specifiche¹⁴. Nega di aver impiegato illecitamente codice per supportare lo sviluppo di Linux e di aver violato qualsiasi segreto commerciale. Rincarica la dose sottolineando che, oltre alla diffusa carenza di informazioni circostanziate in numerosi punti dell'istanza di SCO, Unix è una specifica standard e che il suo marchio identifica precisi sistemi operativi. Nel preambolo di questa risposta si può trovare l'affermazione da parte di IBM secondo cui avrebbe il diritto "regolarmente acquistato, perpetuo e irrevocabile" ad usare il "software proprietario" che SCO ritiene invece abbia utilizzato illecitamente.

14 maggio: Con una lettera inviata ai propri clienti, SCO annuncia la sospensione della vendita di Linux, giudicato "un derivato non autorizzato di Unix"¹⁵. L'annuncio, affidato al general manager Chris Sontag, aggiunge che l'azienda continuerà a supportare i clienti che hanno acquistato soluzioni SCO Linux e Caldera OpenLinux. Nella stessa data, Darl McBride invia una lettera alle aziende facenti parte del "TeamSCO Partner Program" in cui annuncia anche che, in giornata, circa 1500 utenti di Linux dovrebbero ricevere una lettera in cui vengono avvertiti che la responsabilità legale per l'uso di Linux potrebbe estendersi agli utenti finali¹⁶. Lo stesso giorno, Chris Sontag offre per la prima volta la possibilità di visio-

¹³http://www.sco.com/ibmlawsuit/ibm_notice_of_removal.qxd.pdf

¹⁴Defendant International Business Machine answer to the complaint of plaintiff Caldera System, Inc. D/B/A The Sco Group, http://www.sco.com/ibmlawsuit/ibm_response_to_sco_group_complaint_on_april30_2003.pdf

¹⁵SCO Suspends Distribution of Linux Pending Intellectual Property Clarification, <http://ir.sco.com/ReleaseDetail.cfm?ReleaseID=109149>

¹⁶SCO threatens Linux vendors and commercial users, <http://www.linuxdevices.com/news/NS5672501140.html>, punto 1

nare quelle parti di sorgente di Linux che, secondo SCO, violerebbero i suoi diritti¹⁷. Questa possibilità è però condizionata alla firma di un accordo di non divulgazione, accordo il cui intero testo verrà pubblicato da *Linux Journal* qualche settimana dopo¹⁸.

15 maggio: Andrew Josey, per The Open Group, dichiara pubblicamente¹⁹ che SCO possiede soltanto i diritti sul codice sorgente di Unix, mentre i marchi Unix ed Unixware sono di proprietà di The Open Group.

16 maggio: Un gruppo di utenti Linux inizia una raccolta di firme “Hey SCO, denunciarmi!”. Dal sito web dell’iniziativa: “Sono un utente Linux. Credo che le tattiche di SCO contro il sistema operativo scelto da me sono ingiuste, errate e bizzarre. Vorrei essere denunciato perché sono certo che le tattiche di SCO contro Linux falliranno. Ho pubblicato il mio indirizzo di posta elettronica come parte di questa petizione in modo che i legali di SCO possano scrivermi e iniziare le procedure per denunciarmi”. Darl McBride risponderà: “Le azioni di SCO potranno rivelarsi impopolari presso quanti vogliono ricevere vantaggi o benefici dal fatto che Linux sia un sistema libero da usare per le applicazioni professionali”.²⁰

18 maggio: SCO dichiara di non voler perseguire gli utenti Linux. Intanto Microsoft acquista da SCO una licenza d’uso di Unix per la somma tra gli otto e dieci milioni di dollari (c’è discordanza su questo punto)²¹.

¹⁷SCO to Reveal Allegedly Copied Code, <http://www.linuxjournal.com/article.php?sid=6877>

¹⁸SCO NDA Offers Little Information, Much Risk, <http://www.linuxjournal.com/article.php?sid=6923>

¹⁹SCO the “owner” of the UNIX operating system, <http://lwn.net/Articles/33176/>

²⁰<http://www.vnunet.com/News/1140944>

²¹Microsoft to license Unix code, <http://news.com.com/2100-1016-1007528.html>

20 maggio: Caldera International Inc. cambia ufficialmente nome in The SCO Group Inc²².

21 maggio: A Salt Lake City, presso l'archivio legale dello Stato dello Utah, vengono distrutti tre milioni di pagine relative a Microsoft. Si tratta di documenti che erano stati usati da Caldera contro l'azienda di Bill Gates nella causa per comportamento anti-competitivo. Ufficialmente era dal 2000 che SCO doveva sborsare 1.500 dollari al mese per la conservazione di quei documenti, e nell'ottobre precedente aveva richiesto e ottenuto dal giudice distrettuale Dee Benson l'autorizzazione a liberarsene²³. La distruzione avviene alcuni giorni dopo la circolazione della notizia secondo cui Microsoft avrebbe finanziato SCO²⁴.

23 maggio: Dal LinuxTag arriva il contrattacco a SCO. Con una lettera in cui chiede esplicitamente all'azienda statunitense "di dimostrare o di smetterla" con le sue accuse, l'organizzazione non profit tedesca ipotizza l'esistenza di pratiche anti-competitive dietro le asserzioni di SCO²⁵.

28 maggio: Novell dichiara l'intenzione di proseguire le proprie attività su piattaforma Linux²⁶ ribadendo la piena titolarità del diritto d'autore e dei brevetti su Unix SystemV. Novell pubblicherà successivamente anche l'elenco completo delle licenze applicate alle diverse versioni del sistema contestato e il carteggio intercorso con SCO dal 12 maggio 2003 ai primi mesi del 2004²⁷. Vie-

²² <http://ir.sco.com/ReleaseDetail.cfm?ReleaseID=109507>

²³ Microsoft Documents to be Shredded in Utah, Microsoft Documents Finally in Shredder <http://tv.ksl.com/index.php?nid=5&sid=29450>

²⁴ <http://online.wsj.com/article/0,,SB105329732841072600.djm,00.html>

²⁵ LinuxTag challenges SCO, <http://www.theinquirer.net/?article=9679>

²⁶ Novell sfida la posizione di SCO e reitera il suo supporto a Linux, <http://www.novell.com/it-it/news/press/archive/2003/05/pr03033.html>

²⁷ Diritti legali unici di Novell, <http://www.novell.com/it-it/licensing/indemnity/legal.html>

ne reso disponibile sul web anche il carteggio completo tra SCO e Novell²⁸.

29 maggio: In base a una dichiarazione di SCO, se nessuna azienda che attualmente utilizza Linux provvederà a regolarizzare, a pagare per la presunta violazione sarà Linus Torvalds²⁹. Il creatore del sistema operativo libero ribatte che il suo progetto, oltre a essere inizialmente gestito in modo caotico a causa dell'iniziale inesperienza personale, è stato scritto da zero e, così come SCO non può avanzare alcun diritto su BSD realizzato a Berkeley, presso l'Università della California, non può farlo neppure con Linux³⁰.

30 maggio: Darl McBride suggerisce che un modo per concludere l'intera storia sarebbe l'acquisizione della stessa SCO da parte di un consorzio guidato da IBM.³¹

1 giugno: Prima scadenza non rispettata di mostrare il codice che violerebbe i diritti di SCO, la data era stata proposta da Chris Sontag quindici giorni prima, quando aveva invitato un gruppo indipendente a visionare il codice SCO.³²

5 giugno: Nonostante la dichiarazione di venti giorni prima, prosegue la vendita di SCO OpenLinux – malgrado il fatto che il prodotto violerebbe la proprietà intellettuale della stessa SCO e quest'ultima ne avesse già annunciato il ritiro dal commercio³³.

²⁸SCOvsIBMNovellLetters, <http://twiki.iwethy.org/twiki/bin/view/Main/SCOvsIBM-NovellLetters>

²⁹<http://www.marketwatch.com/news/story.asp?guid=%7BC408959E-005A-4E93-9006-B32DCD1FCA22%7D&siteid=yhoo>

³⁰Linus Torvalds: SCO Is “Just Too Wrong”, http://www.businessweek.com/magazine/content/04_05/b3868110_mz063.htm

³¹<http://www.computerworld.com/softwaretopics/os/linux/story/0,10801,81709,00.html>

³²<http://www.linuxjournal.com/article.php?sid=6877>

³³Psst! Want to Buy OpenLinux? It's Still Being Sold Online, <http://www.groklaw.net/article.php?story=30>

10 giugno: Il fronte che chiede a SCO di bloccare le pratiche aggressive si estende a un gruppo di aziende polacche, capeggiate da Cyber Service e IT Zone, i cui servizi sono basati su Linux. Se SCO non dovesse desistere, la minacciano di presentare una contro-denuncia per comportamento anti-competitivo³⁴. Lo stesso giorno Linus Torvalds dichiara di aver chiesto di vedere le linee di codice, ma aggiunge di non essere disponibile a firmare alcun accordo di non divulgazione, come chiesto invece da Darl McBride³⁵. Contemporaneamente si sparge la voce che SCO avrebbe incorporato codice Linux nella propria versione di Unix, in violazione della GPL, nella scrittura del sotto-sistema LKP (Linux Kernel Personality), che serve per far funzionare i programmi per Linux su Unix in forma binaria senza bisogno di ricompilazione³⁶.

12 giugno: “Dubbi irresponsabili”. Sono le parole di Eben Moglen, legale della Free Software Foundation, nell’articolo intitolato “SCO vs. IBM” in cui spiega la confusione dimostrata dai portavoce di SCO nell’assimilare Linux a tutto il software libero senza porre alcuna distinzione tra il sistema operativo e altri programmi di largo utilizzo³⁷. Contemporaneamente SCO diffonde il proprio rapporto finanziario relativo al primo trimestre 2003: si sostiene, tra l’altro, che non necessariamente gli accordi di licenza con Microsoft devono essere rinnovati, che i profitti del perio-

³⁴Polish Linux firms ask SCO to stop ‘unfair practices’,

<http://www.theage.com.au/articles/2003/06/10/1055010962786.html>

³⁵Torvalds Says SCO Won't Give Him Peek At Code Without NDA,

<http://www.crn.com/sections/BreakingNews/dailyarchives.asp?ArticleID=42539>

³⁶Did SCO Violate the GPL?,

http://www.eweek.com/print_article/0,3668,a=43186,00.asp; Cyber Cynic: SCO's Hands in the Source Jar, <http://www.practical-tech.com/business/b06122003.htm>

³⁷SCO v. IBM, http://www.linuxuser.co.uk/articles/lud_special-free_software_matters.html

do in esame sono da imputare solo alle licenze, ma soprattutto che qualsiasi sistema operativo derivato da Unix è di proprietà SCO³⁸.

15 giugno: In un incontro pubblico Darl McBride mostra due esempi di codice che violerebbero il copyright di SCO³⁹. Entro 12 ore gli spezzoni di codice vengono analizzati e sconfessati. McBride replicherà che si era trattato di una dimostrazione esemplificativa, non di una violazione concreta.

16 giugno: SCO annuncia il decadimento immediato per IBM dei diritti d'uso e di distribuzione di AIX e dei file ad esso correlati⁴⁰. Lo stesso giorno arriva la notizia della convocazione dell'udienza preliminare per il 13 agosto alle 9.30, e nel frattempo l'azienda pubblica l'istanza emendata in cui accusa IBM di non aver collaborato nella risoluzione della contesa – pur puntualizzando che la presunta violazione non sarebbe un plagio, ma piuttosto una “derivazione impropria”⁴¹. Nella stessa istanza emendata, SCO triplica la richiesta di danni, portandola da uno a tre miliardi di dollari, così suddivisi: un miliardo per violazione da parte di IBM del contratto con SCO, un miliardo per violazione da parte di Sequent (acquisita da IBM nel 1999) di un altro contratto ed un miliardo per concorrenza sleale⁴².

25 giugno: Sulla vicenda SCO prende la parola anche Bill Gates sottolineando, di fronte agli analisti finanziari di Microsoft, la

³⁸Quarterly report pursuant to section 13,

http://www.sec.gov/Archives/edgar/data/1102542/000110465903012299/j2045_10q.htm

³⁹<http://forum.golem.de/phorum/read.php?f=44&i=1869&t=1716>

⁴⁰SCO Announces Immediate Termination of IBM's Right to Use and Distribute AIX Software and Files for Permanent Injunction,

<http://ir.sco.com/ReleaseDetail.cfm?ReleaseID=111534>

⁴¹Amended complaint, <http://www.sco.com/ibmlawsuit/amendedcomplaintjune16.html>

⁴²SCO suit now seeks \$3 billion from IBM, http://news.com.com/2100-1016_3-1017965.html

“debolezza intrinseca” di Linux. Questo non sarebbe in grado di generare profitti per le aziende, e di garantire la necessità, per chi investe in ricerca e sviluppo, delle sicurezze in materia di proprietà intellettuale⁴³. Due giorni più tardi la FFII (Foundation for a Free Information Infrastructure) contesta le dichiarazioni del patron di Redmond secondo cui, a denti stretti, sosterebbe che il software libero contiene codice Microsoft⁴⁴.

28 giugno: Viene dato avvio alla stesura del documento “Halloween IX: It Ain’t Necessarily SCO”, a cura di Eric Raymond e Rob Landley. Obiettivo principale del lavoro è di raccogliere “menzogne, deliberate distorsioni e inganni” contenuti nella causa SCO. Il documento verrà progressivamente aggiornato (la revisione 1.3 è del 28 agosto 2003) e alla sua stesura lavoreranno complessivamente una trentina di persone⁴⁵.

8 luglio: Nella presentazione dei documenti alla SEC, SCO presenta in modo ingannevole i diritti proprietari su Unix, sostenendo tra l’altro: “Uno dei beni patrimoniali acquisiti da Tarantella riguardava tutti i diritti, titoli ed interessi in e verso Unix e UnixWare, ivi incluso il codice sorgente e i diritti di proprietà intellettuale⁴⁶”.

18 luglio: Si diffonde la notizia che SCO stia lavorando ad un nuovo programma di vendita di licenze Linux: chi le acquisterà eviterà le conseguenze dell’azione legale intentata contro IBM⁴⁷.

⁴³Gates Says SCO’s Case Against IBM Will Harm Linux’s Commercial Prospects, <http://www.crn.com/sections/BreakingNews/dailyarchives.asp?ArticleID=43532>

⁴⁴Gates: Linux infringing numerous patents, SCO is just the beginning, <http://swpat.ffii.org/news/03/index.en.html#crn-gates030725>

⁴⁵Halloween IX: It Ain’t Necessarily SCO, <http://www.opensource.org/halloween/halloween9.html>

⁴⁶<http://www.sec.gov/Archives/edgar/data/1102542/000104746903031063/a2118764zs-3a.htm>

⁴⁷SCO readies new Linux licensing program, http://www.infoworld.com/article/03/07/18/HNscolicense_1.html

21 luglio: Prosegue fino a questa data la diffusione commerciale della distribuzione SCO di Linux, inclusiva delle componenti contestate dalla stessa SCO. Nella stessa giornata, l'azienda rende noto che lo U.S. Copyright Office ha registrato a suo favore il codice sorgente di SystemV, prerequisito per irrigidire i diritti SCO su Unix. Alla battaglia giudiziaria si aggiunge anche RedHat. Il CEO Matthew Szulik, presente al LinuxWorld Conference & Expo, dichiara di essere ricorso alle vie legali allo scopo di dimostrare che non esiste alcuna violazione di proprietà intellettuale né di segreto commerciale⁴⁸.

31 luglio: Eben Moglen pubblica, per conto degli Open Source Development Labs, una ricerca in cui prende posizione sul caso affermando che SCO rende impossibile qualunque risposta coerente alle accuse avanzate, dal momento che queste non contengono elementi fattuali. Sostiene inoltre che gli utenti finali di Linux sono comunque al riparo da qualunque conseguenza della causa legale, esattamente come i lettori di un libro che viola il diritto d'autore non hanno responsabilità riguardo alla violazione. Prende infine posizione sull'offerta di SCO di una licenza d'uso di Linux, dal momento che il sistema open source è ancora liberamente disponibile per il download dallo stesso sito di SCO⁴⁹. Opinder Bawa, vice-presidente anziano dell'unità Engineering and Global Services di SCO abbandona la società⁵⁰.

4 agosto: Alla battaglia legale si aggiunge anche RedHat. Il CEO Matthew Szulik, presente al LinuxWorld Conference & Expo, dichiara di aver avviato l'iter legale allo scopo di dimostrare che

⁴⁸SCO Registers UNIX Copyrights and Offers UNIX License,

<http://ir.sco.com/ReleaseDetail.cfm?ReleaseID=114170>

⁴⁹Questioning SCO: A Hard Look at Nebulous Claims,

http://www.osdl.org/docs/osdl_eben_moglen_position_paper.pdf

⁵⁰<http://twiki.iwethery.org/twiki/bin/view/Main/OpinderBawa>

non esiste alcuna violazione di proprietà intellettuale o di segreto commerciale⁵¹.

5 agosto: Un comunicato-stampa annuncia la SCO Intellectual Property License⁵² per Linux, relativa al sistema operativo che, proprio secondo l'istanza legale, viola i diritti della stessa SCO a partire dalle versioni 2.4 e 2.5 del kernel. Il costo della licenza viene fissato a 699 dollari per singola CPU ed entrerà in vigore a partire dal primo ottobre successivo. Per CPU aggiuntive o sistemi multipli, embedded e desktop saranno adottate tariffe ad hoc. Va aggiunto che il prezzo di 699 dollari è per i server. Per i desktop viene indicato un prezzo di 199 dollari. Viene anche definito un periodo "introduttivo", con scadenza il 15 ottobre, dopo il quale il prezzo per singola CPU per i server si alzerà a 1399 dollari⁵³.

6 agosto: IBM presenta la propria memoria difensiva, replicando alle specifiche contestazioni di SCO⁵⁴. In aggiunta a quanto la società ha già esplicitato nel rigettare le accuse, il documento sottolinea anche la "campagna di falsa pubblicità" da parte di SCO, il cui effetto è la "mistificazione" del mercato Unix e Linux.

13 agosto: SCO dichiara decaduto anche il contratto per Sequent Computer Systems, di proprietà di IBM⁵⁵ per "trasferimento improprio del codice sorgente e dei metodi di sviluppo di Unix Sequent all'interno di Linux". Secondo SCO, sarebbero 148 i file finiti nel kernel di Linux attraverso l'intervento di IBM.

⁵¹Red Hat files suit against SCO, <http://news.com.com/2100-7252-5059547.html>

⁵²SCO Announces Intellectual Property License for Linux, <http://ir.sco.com/Release-Detail.cfm?ReleaseID=115527>

⁵³SCO sets Linux licensing prices, <http://news.com.com/2100-1001-5060134.html>

⁵⁴The text of IBM's counterclaims, <http://lwn.net/Articles/43592/>

⁵⁵The SCO Group Announces Final Termination of IBM / Sequent's Contract to Use or License Dynix Software, <http://ir.sco.com/ReleaseDetail.cfm?ReleaseID=115995>

15 agosto: Mark Heise, uno degli avvocati di SCO, rivela che uno dei punti cardine della causa legale sarà il tentativo di dimostrare l'invalidità della GNU/GPL perché, a suo dire, le clausole che permettono la copia e la modifica senza limitazioni sono incompatibili con la legge statunitense sul diritto d'autore. Posizione immediatamente contestata da Eben Moglen e Eric Raymond⁵⁶.

19 agosto: Vengono alla luce alcuni stralci da una presentazione di Darl McBride relativa alla rivelazione di alcune parti del codice che sarebbe stato copiato da Unix in Linux. In breve, viene così dimostrato come queste parti di codice fossero decadute nel pubblico dominio fin dal 1979⁵⁷.

22 agosto: Con una lettera aperta a Darl McBride, Eric Raymond lo accusa di volersi appropriare del lavoro dell'intera comunità di sviluppo, tentando di distruggere non solo la licenza GNU/GPL, ma tutte le licenze riconosciute dalla Open Source Initiative⁵⁸.

26 agosto: IBM cita in giudizio Canopy Group e Ralph Yarro, richiedendo tra l'altro la presentazione di "tutti i documenti concernenti le evidenze accusatorie della presenza di UNIX in Linux"⁵⁹.

2 settembre: SCO Germania viene condannata da una corte tedesca al pagamento di 10 mila euro a causa delle proprie rivendicazioni sulla proprietà intellettuale⁶⁰. IBM, con una comunicazione

⁵⁶SCO vs. GPL: Luminaries Decry Legal Maneuver as "Posturing", <http://linuxtoday.com/infrastructure/2003081502226NWCYLL>

⁵⁷Analysis of SCO's Las Vegas Slide Show, <http://perens.com/Articles/SCO/SCOSlideShow.html>

⁵⁸An Open Letter to Darl McBride, <http://www.newsforge.com/business/03/08/22/1746248.shtml?tid=19>

⁵⁹<http://twiki.iwethey.org/twiki/bin/view/Main/CanopyGroup>

⁶⁰SCO muss Ordnungsgeld zahlen, <http://www.heise.de/newsticker/meldung/39964>

interna, rincara il contrattacco definendo “FUD⁶¹, non fatti, l’obiettivo dell’attacco di SCO” e accusa l’azienda di aver violato la licenza GNU/GPL⁶².

9 settembre: Darl McBride pubblica la “Open Letter to the Open Source Community”⁶³.

26 settembre: IBM presenta un’altra memoria legale contro SCO, sostenendo che quest’ultima avrebbe violato la GNU/GPL e il copyright di Big Blue facendo credere di possedere i diritti d’autore sul codice di Linux.

30 settembre: Il giudice Kimball accorda a SCO una proroga fino al 4 febbraio 2004 per aggiungere elementi alla propria denuncia legale. In caso contrario, l’istanza verrà archiviata⁶⁴.

1 ottobre: Anche Silicon Graphics viene coinvolta nell’affaire: un comunicato di SCO ne dichiara decaduto il contratto di utilizzo di System V per violazione dei termini della licenza⁶⁵. Nella stessa giornata, la replica di Silicon Graphics spiega di aver immediatamente trovato tre alternative agli elementi contestati: `ate_utils.c`, la funzione `atoi` e il file di header `systeminfo.h`⁶⁶.

17 ottobre: L’agenzia di venture capital BayStar Capital annuncia un finanziamento di 50 milioni di dollari a SCO. Ufficialmente l’iniezione di liquidi andrà a sostenere sia lo sviluppo del

⁶¹Fear, Uncertainty, Doubt: *paura, incertezza e dubbio, tipico acronimo per indicare qualsiasi tipo di disinformazione usata come arma anti-competitiva.*

⁶²IBM Letter to Internal Sales Force, <http://www.eweek.com/article2/0,3959,1212374,00.asp>

⁶³<http://www.caldera.com/company/openletter/>

⁶⁴SCO v. IBM, http://en.wikipedia.org/wiki/SCO_v._IBM_Linux_lawsuit

⁶⁵SCO Turns Its Attention to SGI,

<http://www.cbronline.com/latestnews/c87ea1f4f478527880256db20018c17e>

⁶⁶To the Linux Community, http://oss.sgi.com/letter_100103.txt

software aziendale che le spese legali⁶⁷. Si vedrà più avanti, con la pubblicazione del documento “Halloween X: Follow The Money”, come la questione finanziaria sia ben diversa.

24 ottobre: SCO presenta al tribunale distrettuale dello Utah una richiesta di rigetto della causa intentata da IBM e attacca nuovamente la GNU/GPL definendola “incompatibile con le leggi sul diritto d’autore e sull’Antitrust, le leggi sul controllo delle esportazioni e la Costituzione degli Stati Uniti”⁶⁸.

11 novembre: Si diffonde la notizia di una specifica richiesta, da parte di IBM, alle varie aziende che hanno finanziato SCO o si sono sbilanciate pubblicamente a suo favore nei mesi precedenti. Tra queste, Baystar Capital, Deutsche Bank, Renaissance Ventures e Yankee Group. Si chiede cioè di presentare tutti i documenti in loro possesso riguardanti SCO e il maggior azionista Canopy Group⁶⁹.

4 novembre: Novell trova l’accordo per l’acquisizione del distributore tedesco SuSE, ampliando la propria presenza in ambito Linux dopo aver inglobato anche Ximian di Miguel De Icaza.

12 novembre: SCO risponde all’attacco di IBM richiedendo al tribunale dello Utah di emettere mandati di comparizione nei confronti di sei soggetti, tra singoli ed organizzazioni, tra cui Linus Torvalds e Richard Stallman⁷⁰. Gli Open Source Development

⁶⁷SCO accepts \$50m investment, <http://news.zdnet.co.uk/software/linux-nix/0,39020390,39117195,00.htm>

⁶⁸SCO asks judge to void Linux general public license, <http://www.sltrib.com/2003/Oct/10292003/business/106360.asp>

⁶⁹IBM Subpoenas SCO Investors, Analysts, http://www.forbes.com/technology/enterprisetech/2003/11/11/cz_dl_1111ibm.html

⁷⁰Torvalds hit with SCO Linux lawsuit subpoena, <http://www.silicon.com/management/government/0,39024677,39116876,00.htm>

Labs annunciano che copriranno tutte le spese legali di Torvalds e di qualunque altro dipendente dovesse essere citato in giudizio⁷¹.

24 novembre: L'Open Source Development Labs (OSDL) pubblica una ricerca firmata da Eben Moglen nella quale si mette in dubbio la fondatezza delle istanze di SCO contro la GNU/GPL e si rassicurano gli utenti di Linux circa la legittimità del codice che, secondo l'accusa, sarebbe stato copiato⁷².

5 dicembre: Le autorità inquirenti ordinano a SCO di presentare la porzione di codice sotto accusa prima che lo faccia IBM. Le viene concessa una finestra di trenta giorni per provvedere⁷³.

22 dicembre: SCO annuncia ulteriori iniziative per rafforzare i propri diritti sul codice di SystemV (certificazioni scritte di non violazione della licenza, accordi di non divulgazione). E tira in ballo, per la prima volta, anche presunte infrazioni al Digital Millennium Copyright Act (DMCA), inviando un migliaio di lettere ad altrettanti utenti di Linux⁷⁴. In realtà, la lettera è datata 19 dicembre⁷⁵, anche se effettivamente è stata resa pubblica solo il 22. Per la prima volta compare una lista ben precisa di file Linux che, secondo SCO, conterrebbero codice di sua proprietà. Il giorno stesso, Linus Torvalds risponde che i file indicati nel documento sono stati scritti personalmente da lui.

⁷¹OSDL to fund Linus Torvalds legal representation in SCO litigation, http://www.osdl.org/newsroom/press_releases/2003/2003_11_14_beaverton.html

⁷²SCO: without fear and without research, http://www.osdl.org/docs/osdl_eben_moglen_second_statement.pdf

⁷³La causa tra SCO e IBM, http://www.softwarelibero.it/progetti/scosaga/sco_saga-1.3.html

⁷⁴SCO Announces New Initiatives to Enforce Intellectual Property Rights, <http://ir.sco.com/ReleaseDetail.cfm?ReleaseID=125089>

⁷⁵http://www.sco.com/scosource/abi_files_letter_20031219.pdf

12 gennaio 2004: Dopo l'udienza del 5 dicembre in cui era stato imposto a SCO di presentare entro trenta giorni il codice incriminato, l'azienda torna davanti al giudice senza aver adempiuto a tale mandato. Intel crea un fondo per le spese legali a difesa di Linux.

13 gennaio: Novell si aggiunge a Intel nella raccolta-fondi per Linux e rilascia un dossier di corrispondenza con SCO⁷⁶.

14 gennaio: SCO presenta il piano di vendita licenze anche nel resto del mondo. Chris Sontag afferma che, nel giro di poche settimane, SCO comincerà a denunciare gli utenti finali di Linux⁷⁷.

29 gennaio: Sono 350 mila i computer in tutto il mondo che vengono infettati da un nuovo worm, Mydoom, e relativa variante, Mydoom.B. Come conseguenza, nello scenario tipico di un attacco di diniego di servizio distribuito (Distributed Denial of Service, DDoS), il 20 per cento del traffico web globale viene rediretto contro i siti web di Microsoft e di SCO. Quest'ultima offre una 'taglia' di 250mila dollari a chi darà informazioni utili alla cattura degli autori e avanza il sospetto che l'attacco arrivi dalla comunità open source, suscitando una marea di proteste e smentite⁷⁸.

6 febbraio: Si svolge l'udienza già prevista per il 23 gennaio, rinviata per dare la possibilità a SCO di completare la documentazione da sottoporre al giudice. Di fatto, non viene però presentato alcun elemento che metta in dubbio la legittimità del codice di Linux, mentre IBM lamenta la mancanza di risposte alle sollecitazioni delle settimane precedenti.

⁷⁶<http://twiki.iwethey.org/twiki/bin/view/Main/SCOvsIBMNNovellLetters>

⁷⁷SCO brings Linux licensing to Europe, <http://news.zdnet.co.uk/software/0,39020381,39119087,00.htm>

⁷⁸Mydoom becomes the fastest-growing e-mail virus ever, http://www.liberaltopia.org/archives/2004_01.html

9 febbraio: SCO lascia cadere la violazione del segreto commerciale, ma rinforza la richiesta di danni a IBM portandola a cinque miliardi di dollari⁷⁹. Tre giorni prima aveva citato in giudizio ancora una volta la stessa IBM per il mancato rispetto degli accordi contrattuali⁸⁰.

18 febbraio: SCO chiama in causa AutoZone, società con base a Memphis (Tennessee) che vende accessori e software diagnostici automobilistici. L'accusa: nel 2002, dopo essere stata cliente del querelante, AutoZone era passata a Linux avvalendosi di RedHat e IBM. In questo caso, la presunta violazione riguarderebbe alcune librerie SCO che, afferma Jim Greer, ricercatore nell'ambito dell'intelligenza artificiale che ha collaborato ai processi di transizione di AutoZone, non sarebbero rientrate nel porting effettuato⁸¹.

3 marzo: Viene diffuso il documento "Halloween X: Follow The Money", in cui si ricostruisce il percorso di 68,5 milioni di dollari, trasferiti da Microsoft verso SCO, a titolo di finanziamento per le spese legali⁸². Nello stesso giorno, ancora SCO querela un'altra azienda automobilistica, la Daimler-Chrysler: in ballo sempre il System V, il cui utilizzo non avrebbe rispettato le modalità contrattuali⁸³.

(Si ringrazia Roberto Orsini per la preziosa collaborazione a questa cronologia).

⁷⁹SCO amends complaint against IBM, <http://www.smh.com.au/articles/2004/02/09/1076175080452.html>

⁸⁰SCO Tries To Tack Copyright Infringement To IBM Claim, <http://www.crn.com/sections/BreakingNews/dailyarchives.asp?ArticleID=47787>

⁸¹Exhibit 1 to IBM's Report on SCO's Compliance, <http://www.groklaw.net/article.php?story=20040215015800694#c78161>

⁸²Halloween X: Follow The Money, <http://www.opensource.org/halloween/halloween10.html>

⁸³Complaint and jury demand, <http://news.findlaw.com/hdocs/docs/cyberlaw/scodchrys30304cmp.pdf>

Indice

Prefazione	3
Unix questo sconosciuto	5
In principio era Novell... ..	40
L'attacco di SCO e McBride	81
Le reazioni della comunità del software libero e open source	115
Cronologia	133

direttore editoriale **Marcello Baraghini**

<http://www.stampalternativa.it/>

e-mail: redazione@stampalternativa.it

S
T
A
M
P
A
L
T
E
R
N
A
T
I
V
A

Contro il comune senso del pudore, contro la morale codificata, controcorrente. Questa collana vuole abbattere i muri editoriali che ancora separano e nascondono coloro che non hanno voce. Siano i muri di un carcere o quelli, ancora più inviolabili e resistenti, della vergogna e del conformismo.

**Beccaria, Marchetti-Stasi,
Parrella, Somma (NMI Club)**

Noscopyright

progetto grafico **Anyone!**

impaginazione **Roberta Rossi**

© 2004 **Nuovi Equilibri**

Casella postale 97 - 01100 Viterbo fax 0761.352751

e-mail: ordini@stampalternativa.it

Attenzione! I manoscritti inviati all'editore non si restituiscono.
Non vengono forniti pareri e schede di lettura.
Non si considerano testi inviati per e-mail.

finito di stampare nel mese di aprile 2004

presso la tipografia **Graffiti**
via Catania 8 - 00040 Pavona (Roma)