

# Instant Basic for Java

Java sembrerebbe avere tutte le caratteristiche del linguaggio di programmazione ideale: semplicità, portabilità, possibilità di riutilizzo del codice, affidabilità, sicurezza. Non è detto che per sfruttare tutte queste possibilità sia indispensabile imparare un nuovo linguaggio ma si può fare ricorso a questo strumento che permette la scrittura o la conversione in Java di programmi scritti in un linguaggio diffuso, il Visual Basic.

di Giuseppe Zanetti

Programmare è un'arte difficile, che richiede notevoli investimenti in tempo ed in denaro, sia da parte dell'utente finale, su cui tali costi ricadono, che del programmatore, il quale spesso è costretto ad acquistare tool di sviluppo costosi e imparare ad utilizzarli o a investire nella realizzazione in proprio di librerie, il cui costo dovrebbe poi essere ammortizzato negli anni.

Per questi motivi sarebbe auspicabile che il ciclo di vita di un programma fosse abbastanza lungo ed il suo codice riutilizzabile o aggiornabile velocemente, in modo da preservare nel futuro gli investimenti effettuati.

A prescindere dall'aspetto puramente economico, il ciclo di vita di un programma dovrebbe essere prima di tutto un compromesso fra la necessità di aggiungere nuove funzioni e quella di avere un prodotto il più possibile affidabile. Infatti ad ogni modifica del codice vi è la possibilità di introdurre nuovi errori, anche in parti già stabili.

Per questo vi è la necessità di tutelare il software dell'utente anche nel caso di aggiornamento del sistema operativo. I produttori di sistemi commerciali lo sanno benissimo ed hanno sempre offerto programmi di migrazione alle nuove piattaforme il più indolore possibili. Dato l'enorme parco software sviluppato negli anni, anche i sistemi operativi e le tecnologie emergenti non possono non affrontare il problema, se vogliono riuscire a portare nuovi utenti dalla loro parte.



Logo della Halcyon.

Ne abbiamo già parlato descrivendo il Network Computer di Corel ed in questo articolo, vedremo uno strumento che può facilitare questa migrazione.

## L'utopia del C++

Quando ho frequentato il corso di "Elaborazione Automatica delle Informazioni", sembrava che la risposta definitiva a tutti i problemi, passati e futuri, di portabilità e riusabilità del codice fosse il C++.

Nonostante il C++ non abbia mantenuto completamente le promesse e risolto tutti i problemi, alcune delle idee presenti in esso hanno avuto comunque una grande influenza sul modo di

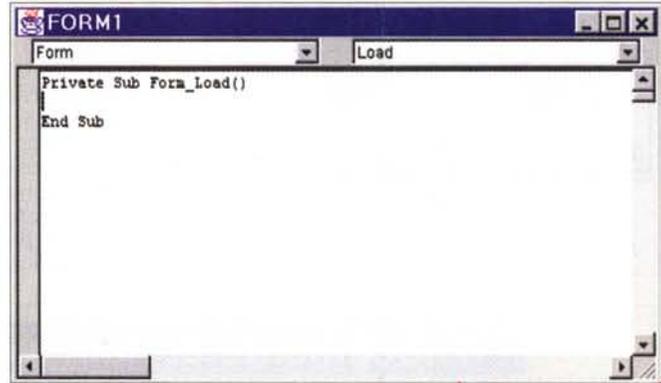
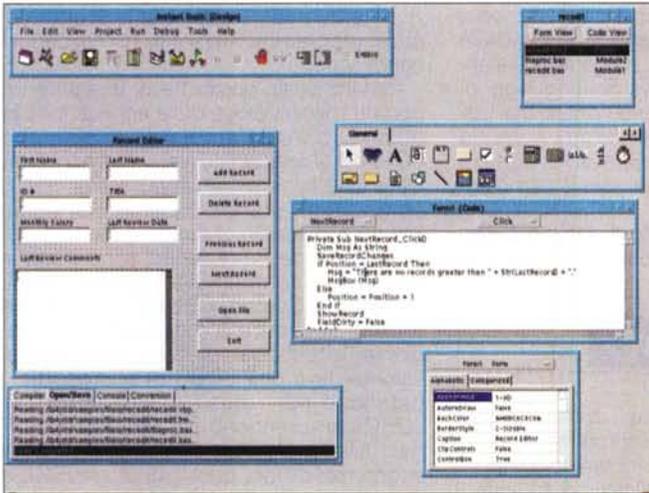
fare programmazione degli anni seguenti.

Il parziale insuccesso del C++ nel mondo UNIX, che ha continuato ad utilizzare come linguaggio di elezione il buon vecchio C, è dovuto non solo a motivi culturali e di tradizione, bensì anche alla minore efficienza del nuovo linguaggio ed alla mancanza di librerie scritte appositamente per esso, fatte forse salve quelle relative alle interfacce grafiche.

Nel mondo Windows invece il C++ ha avuto un maggior successo, ma ha anche dovuto subire la concorrenza di altri linguaggi che nel frattempo si erano convertiti alla filosofia degli oggetti, come Visual Basic, e Delphi, erede del glorioso Turbo Pascal.

La fortuna di questi linguaggi è dovuta soprattutto alla semplicità di programmazione dell'interfaccia grafica, che per l'utente medio di Windows è alle volte più importante rispetto alla funzionalità stessa del prodotto, ed alla possibilità di utilizzare componenti già pronti scritti da altre persone, semplicemente trascinandoli all'interno del proprio programma, settandone in modo opportuno delle proprietà e richiamando dal codice i metodi che ognuno di essi mette a disposizione.

L'approccio visuale, anche se a livelli avanzati dimostra i propri limiti, facilita di molto la fase di apprendimento del linguaggio e rende possibile eseguire un minimo di programmazione in tempi brevi anche da parte di persone alle prime armi.



Una vista d'insieme del prodotto: l'ambiente del Visual Basic è riprodotto nei minimi particolari... anche nelle troppe finestre.

L'editor.

La finestra principale dei comandi.

## Java, il C++ nell'era di Internet

Le aspettative che alcuni anni or sono erano riposte nel C++, ora possono essere riproposte pari pari parlando di Java.

Esso, come linguaggio, e soprattutto come approccio, sembrerebbe avere tutte le caratteristiche del linguaggio di programmazione ideale: semplicità, portabilità, possibilità di riutilizzo del codice, affidabilità, sicurezza.

Queste caratteristiche sono ottenute facendo uso di una macchina virtuale, una specie di emulatore che interpreta il codice oggetto generato dal compilatore (bytecode) in modo indipendente dal sistema che "sta sotto", senza necessità di dover ricompilare il sorgente passando da una architettura all'altra.

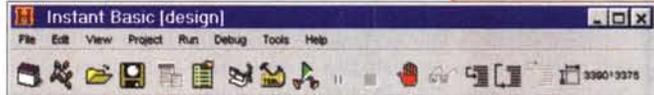
Il linguaggio utilizzato per produrre il bytecode è di solito il Java (si presti attenzione all'omonimia fra il linguaggio e l'intero sistema basato sulla macchina virtuale) tuttavia il codice oggetto può essere ottenuto anche mediante altri strumenti, ad esempio compilatori di altri linguaggi. Ciò non deve essere inteso in modo troppo semplicistico: il linguaggio Java infatti è scritto, oltre che per essere più semplice dei linguaggi precedenti, anche per superare certe ambiguità formali o dettagli non perfettamente specificati, che complicavano la portabilità dei programmi.

Diversi processori e sistemi operativi possono infatti rappresentare i numeri o le strutture di dati con un numero differente di bit, non necessariamente allineandoli in memoria allo stesso modo.

Ciò è importante specialmente quando si lavora in rete, in quanto questi "dettagli" possono essere causa di incommunicabilità fra i diversi sistemi.

Per permettere comunque la comunicazione è possibile applicare varie strategie, come quella di convertire la rappresentazione interna delle strutture di dati in una rappresentazione esterna definita e comune. Tale approccio, oltre a complicare il programma col codice necessario ai controlli, può avere lo svantaggio di appesantire inutilmente i dati che vengono trasmessi in rete.

L'idea di utilizzare una macchina virtuale come interfaccia fra il software e la macchina reale (sistema operativo e hardware), permette di far sì che il pro-



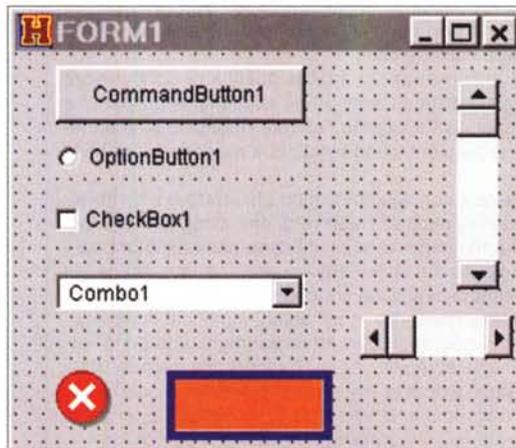
grammatore si trovi davanti sempre il medesimo ambiente con tutte le problematiche precedenti già risolte.

L'emulazione via software della macchina virtuale è un'operazione molto costosa in termini di risorse di calcolo e questo spiega la poca efficienza con cui funzionano i programmi Java nelle macchine attuali. Inoltre, l'implementazione standard della macchina virtuale Java, fornita da Sun nel JDK (Java Development Kit), è scritta ponendo maggiore attenzione alla portabilità rispetto che all'efficienza. Altre case di software producono versioni più efficienti ed iniziano ad apparire sul mercato processori in grado di lavorare direttamente col bytecode.

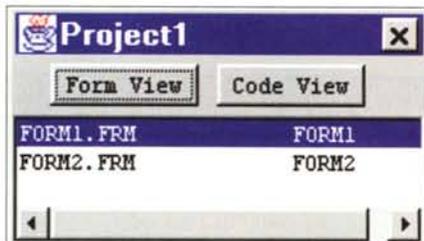
## Instant Basic for Java, ovvero Java senza Java

La possibilità di produrre direttamente bytecode è pienamente utilizzata da Instant Basic for Java, un prodotto della Alcyon Software (<http://www.alcyonsoft.com>) che permette di creare codice Java a partire da sorgenti scritti in Visual Basic.

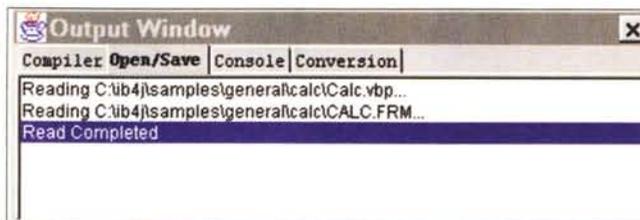
Probabilmente molti lettori storceranno il naso sentendo parlare di "programmazione in Visual Basic", bisogna tuttavia tenere presente che



Lo strumento che permette di disegnare l'interfaccia grafica.



La finestra con la lista dei moduli che compongono il progetto corrente.



La finestra di stato.

esiste una quantità immensa di software scritto in questo linguaggio. Esso deve essere pensato un po' come l'equivalente nel mondo Windows dei linguaggi di scripting di UNIX, spesso limitati ma utili per produrre piccoli programmi in modo rapido e poco costoso. Spesso si tratta di programmi che non è possibile cambiare dall'oggi ai domani.

Lo scopo del prodotto di Alcyon è perciò quello, duplice, di permettere la con-

versione in Java del software esistente e la sua manutenzione e di rendere possibile la scrittura di nuovi programmi mediante un linguaggio conosciuto.

Instant Basic permette di eseguire un porting molto veloce delle applicazioni, in quanto, oltre all'ambiente di sviluppo (IDE) ed al linguaggio vero e proprio, offre anche la versione JavaBeams dei 16 componenti standard del Visual Basic (menù a discesa, checkbox, listbox, ...). Nella versione Professional sono disponibili anche i controlli DAO (Data Control, DBGrid, DBCombo, DBList), che permettono di accedere a database SQL mediante l'interfaccia standard JDBC (l'equivalente Java dell'ODBC di Microsoft). I database per i quali esiste un driver JDBC comprendono DB2, Oracle, Informix, Microsoft SQL, ma anche prodotti economici o free, quali mSQL, MySQL e Postgres.

Poiché tutti i "controlli" vengono forniti come JavaBeams, il codice ottenuto può essere ridistribuito senza dover pagare royalty. Oltre a quelli forniti di serie è possibile creare i propri componenti personali o utilizzare JavaBeam scritti da altri.

Il prodotto della Alcyon non è specifi-

## Il documento di Halloween

La tradizione vuole che nella notte di Halloween tutto possa accadere... anche che un memorandum interno di Microsoft, in cui si spiega cosa sono e come affrontare la concorrenza di Linux e dell'Open Source Software (OSS), finisca nelle mani di Eric S. Raymond (ricordate l'autore di "La Cattedrale ed il Bazaar"?) e venga pubblicato nel sito ufficiale del movimento per l'Open Source per poi essere ripreso dalle maggiori testate internazionali, fra cui The Wall Street Journal, The Washington Post, The Boston Globe, The New York Times, PC Week, CNet, Reuters, CNN, ...

Il documento, scritto nel mese di agosto 1998 da Vinod Valloppillil, ingegnere di Microsoft a cui "di tanto viene affidata la responsabilità di tenere sotto controllo ed analizzare le condizioni del mercato e le offerte dei competitori", è suddiviso in due parti, denominate da Raymond "Halloween I" e "Halloween II", disponibili all'indirizzo <http://www.opensource.org/halloween.html>.

Su <http://www.microsoft.com/ntserver/highlights/editorletter.asp> è invece disponibile la conferma di Microsoft sulla veridicità del documento.

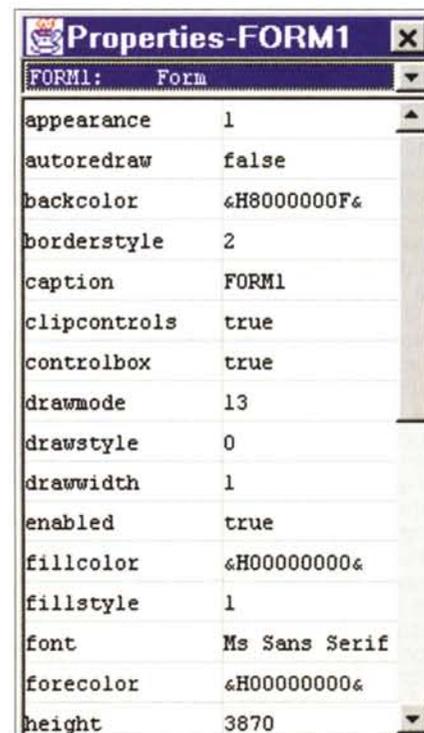
La traduzione in italiano dei documenti e della risposta di Microsoft, a cura di Flavio Romani ([romanif@netbusiness.it](mailto:romanif@netbusiness.it)), si può trovare nel sito della Syntax Error Technology, all'indirizzo <http://se.4net.com/>.

Halloween I esamina il mercato e le tecnologie Open Source attualmente più significative (Linux, Mozilla, Apache, Java, Samba, ...), mentre il secondo documento pone la propria attenzione in particolare su Linux, esaminandone non solo gli aspetti tecnici, bensì anche quelli commerciali, relativi ai prodotti ed alle aziende che vi gravitano attorno (Red Hat, Caldera, Sun, Netscape, ...).

La parte più interessante riguarda l'analisi dei diversi modelli di sviluppo del software adottati dall'azienda leader nel mercato del software e dal movimento che spalleggia l'Open Source Software. La stessa Microsoft riconosce molti dei vantaggi insiti in quest'ultimo: parallelismo intrinseco, possibilità del libero scambio di idee e tecnologie, credibilità a lungo termine, solidità del prodotto, elevata visibilità, capacità del procedimento OSS di radunare e sfruttare l'IQ (coefficiente intellettuale) collettivo di migliaia di individui attraverso Internet. Tutti benefici che non sono replicabili dal modello attuale di produzione di una azienda che tratta software commerciale.

La parte più discussa (e discutibile) del documento riguarda la politica di concorrenza adottabile contro l'OSS, che dovrebbe avere come bersaglio "un procedimento più che una compagnia" e dovrebbe prevedere, secondo Valloppillil, l'utilizzo di protocolli proprietari che aggiungano nuove funzionalità ma nel contempo tendano a rendere più difficile l'integrazione in rete con altri sistemi.

Prima di cadere nella tentazione di giungere a conclusioni troppo affrettate e trionfalistiche, è bene ricordare la posizione attuale dell'azienda di Redmond, che attualmente è accusata dal DOJ, il Dipartimento di Giustizia americano, di aver utilizzato pratiche monopolistiche per annientare la concorrenza nel mercato del software. Accusa che verrebbe a cadere nel caso Microsoft riuscisse a dimostrare di avere dei concorrenti... visto in quest'ottica, lo spirito del documento potrebbe essere riassunto in un unico, significativo, brano, tratto dalla risposta di Microsoft (<http://www.microsoft.com/ntserver/highlights/editorletter.asp>): *The operating system industry is characterized today by vigorous competition. This competition, of which Linux is only a part, exists at the technology level as well as in terms of business models, applications, channels and alliances.* L'industria dei sistemi operativi è oggi caratterizzata da una notevole (sic!) competizione. Questa competizione, di cui Linux è solo una parte, esiste sia a livello tecnico che in termini di modelli di business, applicazioni, canali ed alleanze.



La finestra per modificare le proprietà dei componenti.

Dal 7 al 9 ottobre si è svolto presso la Facoltà di Ingegneria dell'Università La Sapienza di Roma, per il secondo anno consecutivo, il Linux Meeting 1998 (LiMe 98). L'evento tradizionalmente organizzato dal gruppo utenti Linux di Roma (LUG Roma) e dalla IEEE Student Branch si è svolto per questa edizione in collaborazione con il gruppo utenti Linux nazionale (Pluto) diventando LiMe 98 - Pluto Meeting 1998 e riscuotendo una risonanza nazionale.

Gabriele Paciucci (Gabriele.Paciucci@pcainf1.ing.uniroma1.it) mi ha inviato il seguente resoconto. Per ulteriori informazioni: <http://www.pluto.linux.it/meeting98/index.html>

Gli organizzatori hanno voluto dare all'evento un forte carattere di apertura tipico del mondo Linux rendendo gratuito a tutti la partecipazione. La manifestazione è stata finanziata grazie agli sponsor, a tutti gli Exhibitor e ai partner tecnici.

La risonanza e le notorietà che sta avendo in questi mesi Linux ha spinto più di 450 persone a iscriversi a tutte e tre le giornate della conferenza e non si è trattato solamente di studenti tradizionalmente legati al mondo Linux, ma di professionisti del settore informatico in generale e delle telecomunicazioni in particolare. Tutti gli iscritti e i non iscritti, presentatisi in massa il giorno dell'apertura, hanno potuto ricevere la cartellina della manifestazione contenente fra l'altro un CD in cui sono stati raccolti gli atti della conferenza e la distribuzione Debian 2.0 in italiano.

La manifestazione si è svolta nella stupenda cornice del chiostro cinquecentesco della Facoltà di Ingegneria, dove sono stati allestiti gli stand per poter permettere ai partecipanti di conoscere gli operatori commerciali sul sistema operativo Linux. Oltre a RedHat Italia, CD Eurom (SuSE GmbH) e Cofax (Caldera) rivenditori per l'Italia delle maggiori distribuzioni, erano presenti Edizioni Infomedia e Libreria Athena, SciSoft, Jolly Bergamo, Axel, Eurotech, Securtech, Yepa e SeeWeb.

Nella sala delle lauree, attigua al chiostro, si è svolto il cuore della manifestazione. Nelle tre giornate si sono avvicendati ben 26 interventi e due importanti tavole rotonde.

Nella prima giornata si è trattato lo spinoso problema del dualismo sempre più marcato in Linux tra sviluppo Open Source e attività commerciali. Tale discorso porta-

to avanti di volta in volta dai vari interventi con esperienze diverse, ma sempre interessanti, è naturalmente sfociato nella tavola rotonda moderata da Alessandro Rubini e che ha infervorato i presenti. I discorsi più che su "Open Source - Commerciale" sono andati a sfociare sul ruolo che dovrà avere Linux per essere vincente sugli altri sistemi operativi. Alcuni erano dell'idea che era giunto il momento dello scontro frontale sul campo dei desktop, mentre altri più moderati, credevano opportuno che Linux continuasse a prevalere e proliferasse in quelle nicchie di mercato in cui ha maggior penetrazione... al tempo l'ardua sentenza.

La seconda giornata ha avuto un taglio più divulgativo, sfociando nella tavola rotonda riguardante "Linux e la didattica" moderata da Giulio Mazzolini e che aveva l'importante presenza del Dott. Mazza rappresentate del Ministero della pubblica istruzione. Il Dott. Mazza ha garantito l'assoluta imparzialità del Ministero verso l'adozione da parte degli Istituti scolastici di

un qualsiasi sistema operativo. Lo stesso Mazza ha recepito positivamente le proposte fatte riguardo ad un forum permanente e a un informativa fatta attraverso i canali ministeriali per far conoscere e insegnare il sistema operativo Linux.

Nel terzo giorno, infine, si è dato spazio sia agli interventi riguardanti aspetti tecnici, sia ai più importanti progetti italiani che coinvolgono questo incredibile sistema operativo.

Un afflusso eccezionale ha avuto la showroom in cui erano a disposizione dei presenti un certo numero di macchine Intel, due Sparc e alcuni sistemi embedded, tanto da rendere impraticabili alcune dimostrazioni e tanto da dover spostare in un'altra aula le installazioni delle distribuzioni RedHat, SuSE, Caldera e Debian.

Sperando che il desiderio degli organizzatori di far conoscere Linux non tanto come un sistema operativo gratuito, quanto di un sistema operativo LIBERO, sia andato soddisfatto, arriverci alle prossime edizioni del Linux Meeting e del Pluto Meeting.

Gli organizzatori vogliono ringraziare in modo particolare la società SIPR per aver curato l'amministrazione dell'evento e tutte le ragazze della segreteria così volenterose e disponibili.

Meeting Press Bureau



sponibile su <http://www.blackdown.org/java-linux.html>.

L'ambiente di sviluppo (IDE) è molto simile a quello di Visual Basic, con la possibilità di disegnare in modo "visuale" l'interfaccia grafica del programma che si sta scrivendo, gestire progetti, progettare menu a discesa, eseguire il codice all'interno di un debugger. Il bytecode ottenuto dal compilatore interno al prodotto (non viene usato quello del JDK), può essere fatto girare come applicazione "standalone" oppure come applet all'interno delle proprie pagine Web.

Nel caso si decida di compilare il proprio programma come standalone, le nuove versioni del kernel di Linux offrono la possibilità di eseguire direttamente i programmi Java dalla linea di comando, come se si trattasse di programmi nativi. Il kernel si occupa di riconoscere automaticamente il formato del codice oggetto e di richiamare in modo trasparente per l'utente la macchina virtuale del JDK.

Oltre ai controlli per accedere ai database, di cui abbiamo già parlato, la versione Professional offre in più rispetto a quella Standard, uno strumento per generare i programmi di installazione del software ed un convertitore che permette di creare bytecode direttamente da sorgenti Visual Basic, senza dover passare attraverso l'IDE. Essa permette inoltre di collegarsi a JavaBeans o ActiveX funzionanti sulla stessa o su altre macchine della rete.

## Conclusioni

Instant Basic sembra coniugare in sé due richieste importanti dei programmatori: la possibilità di ottenere un prodotto funzionante in modo semplice ed in tempi relativamente brevi (punto di forza dei compilatori visuali) e la necessità di portabilità (la scommessa di Java).

Abbiamo già visto, parlando del network computer di Corel, come Linux possa inserirsi in modo efficace anche nell'architettura Java. A mio vedere, l'accettazione di entrambi questi sistemi da parte dell'utente "medio" dipende, oltre che dalla qualità, anche dalla quantità di software disponibile. In quest'ottica Instant Basic for Java offre la possibilità di eseguire un porting molto veloce delle migliaia di programmi e programmini disponibili in Visual Basic.

I programmi creati con Instant Basic possono funzionare senza alcuna modifica su sistemi operativi diversi ed il fatto che non si tratti di un prodotto progettato in modo specifico per Linux non deve essere visto come un difetto, bensì come un ulteriore vantaggio.

co per Linux, ma, essendo esso stesso scritto in Java, può essere usato su qualunque sistema per cui sia disponibile una macchina virtuale. Fra le piattaforme in cui il programma è stato testato vi sono Windows 95, NT 4.0, Solaris, AIX,

IRIX.

Per far funzionare Instant Basic è necessario prima di tutto installare nella propria macchina il JDK di Sun. Se non lo si è ancora fatto questa può essere l'occasione buona per farlo. Esso è di-

MS