

Programmazione.it v6.0 beta

Ciao, per farti riconoscere devi fare il [login](#). Non ti sei ancora iscritto? Che aspetti, [registrati adesso!](#)



Cerca negli Articoli

Cerca nei Libri

Cerca nel Forum

Hello open source
Good-bye chaos
Introducing JBuilder® 2007

Computer, pessimi generatori di numeri casuali

Scritto da [Alessandro Carichini](#) il 30-11-2006 ore 12:53

Può sembrare un paradosso, ma in realtà, quando si pensa alla casualità, non viene certo in mente l'anima deterministica che è alla base della logica dei computer. Così quando si parla di RNG (Random Number Generator) in realtà si dovrebbe parlare di PRNG (Pseudorandom Number Generator), perché questo genere di algoritmo, per quanto casuale possa essere, mantiene sempre un fattore di periodicità.

I numeri casuali hanno applicazioni in molte aree: simulazione, giochi, crittografia, statistica ed altro ancora. Tutti i linguaggi di programmazione contengono una funzione *random* per questo genere di calcoli e molti si basano ancora sull'algoritmo di Donald E. Knuth, emerito professore dell'Università di Stanford e famoso per il suo "The Art of Computer Programming".

Alcuni linguaggi, oltre ad avere la funzione standard, ne prevedono un'altra più avanzata: nel Perl, ad esempio, abbiamo di default rand() ed è, inoltre, possibile scaricare da CPAN (Comprehensive Perl Archive Network) la Math::TrulyRandom ed usare la funzione truly_randon value().



In PHP, la funzione srand() viene affiancata da mt_srand(), la quale si basa sull'algoritmo Marsenne twister.

Anche il framework .NET prevede due generatori di numeri *random*, il primo è il System.Random ovvero la funzione più generica, mentre il secondo è il System.Security.Cryptography.RandomNumberGenerator una vera e propria suite di classi per la crittografia. Per gli amanti del Visual Basic .NET eccone un'interessante implementazione.

Come dicevo i numeri pseudo-casuali vengono scelti con un'eguale probabilità da un insieme finito di numeri. Il numero scelto, però, non è completamente casuale visto che è il risultato di un algoritmo matematico, ma è sufficientemente *random* per un uso pratico. Non a caso c'è un RFC (Request For Comments) anche se un po' datato, intitolato Randomness Recommendations for Security che non parla di uno standard specifico, ma è una sorta di monito alla comunità di Internet in ambito di sicurezza sull'uso dell'RNG.

A tal proposito c'è chi si chiede se sia mai possibile sbancare un casinò grazie al Computer. La risposta ce l'ha data Kevin Mitnick nel suo ultimo libro "L'arte dell'intrusione", il quale, nel primo capitolo, parla di un gruppo di programmatori che riuscirono a vincere ai casinò di Las Vegas un milione di dollari. Tutta la truffa si basava sulla scoperta di una frettolosa implementazione RNG a 32 bit presente nel firmware delle slot machine e videopocker dei primi anni '90. Infatti una volta capito che il sistema *random* era tutt'altro che *pseudo*, ed anzi *ultra deterministico*, scrissero un algoritmo che, a seconda della sequenza di carte che apparivano sul video, ne poteva prevedere le successive.

In definitiva i computer sono dei magnifici *masticatori* di numeri, ma dei pessimi generatori di numeri casuali; se non ne siete convinti, questo [articolo](#) fa al caso vostro!



[Commenta](#)

Precedente: [Maggiore protezione a phpinfo](#)

Successiva: [Aerei senza pilota e un pilota per più aerei](#)

Intervento di [Diego De Zan](#) a.k.a. [simulacron](#) del 11-12-2006 ore 19:11, Pinerolo (TO)



Cavaliere
(124 interventi)
Iscritto il 13-07-2005

Il principale problema dei 'puter è che hanno dei limiti nei numeri che usano e la generazione casuale ne risente....



[Rispondi](#)

Programmazione.it v6.0 beta

Ciao, per farti riconoscere devi fare il [login](#). Non ti sei ancora iscritto? Che aspetti, [registrati adesso!](#)



Cerca negli Articoli

Cerca nei Libri

Cerca nel Forum

Introducing JBuilder® 2007

**Add RAD advantages to
your Eclipse development.**

L'arte della programmazione in Perl

Scritto da [Alessandro Carichini](#) il 09-11-2006 ore 17:35

Avete letto ultimamente un interessante [tutorial](#) per imparare il Perl e oramai vi sentite dei programmatori provetti? Ricordate che per esserlo non basta scaricare dal [CPAN](#) le librerie più *esotiche* per poi fare un brutale copia e incolla degli esempi o degli [snippet](#) di altri sul proprio codice. Programmare è una cosa seria, occorre mettersi di buona lena per evitare che quella che viene considerata come una flessibilità del linguaggio, non diventi un'arma a doppio taglio.

Ma andiamo con ordine e cominciamo con lo scrivere codice in modo *serio* seguendo alcuni concetti chiave presi da [questo articolo](#), il cui inciso recita: *The absolute minimum you need to know.*

Prima però una piccola precisazione: nel Perl coesistono due [sistemi](#) simultanei (per ragioni di retro-compatibilità), che gestiscono lo *spazio dei nomi*, oppure, per i puristi, [namespaces](#). Fino alla versione 4 quando si definiva una funzione o una variabile all'interno di un *package* era visibile da tutti i sotto *package* senza utilizzare alcun *qualificatore* che ne indicasse la provenienza. Grazie al cielo dalla versione 5 le cose si sono sistemate, peccato però che gran parte del codice in circolazione negli script e negli esempi (soprattutto in certi tutorial datati) è rimasto fermo alla versione 3 generando dei [cattivi maestri](#).

Le [classi](#) e gli [oggetti](#) in Perl vengono visti dai più come un qualcosa di ineluttabile quando li si usa da moduli esterni (da CPAN appunto) ed impraticabile quando siamo noi a doverne fare uso. Una [piccola guida](#) per chiarire la faccenda e cominciare ad usarle come delle vere strutture del proprio codice.

Infine, una volta che avete capito come rendere perfetto il vostro codice, è necessario documentarlo e qui entra in scena il buon vecchio [POD \(Plain Old Documentation\)](#), ovvero la cara e vecchia documentazione testuale senza inutili fronzoli. Una sorta di [man pages](#) di Unix.

Questo [POD in 5 minutes](#), che sembra la ricetta della *torta della nonna*, vi insegnerà a documentare con poco sforzo ed un'alta resa il vostro codice.

Come dicevo all'inizio, l'estrema flessibilità della sintassi Perl rende tanto semplice scrivere codice quanto difficoltoso poi diviene da leggere e mantenere per cui è necessario mantenere una ferrea metodologia di lavoro.



[Commenta](#)

Precedente: [Le prime prove con Fedora Core 6 \(1/2\)](#)

Successiva: [Oracle Database 10g, migliori prestazioni anche con grandi basi di dati](#)

Copyright Programmazione.it™ 1999-2006. Alcuni diritti riservati. Testata giornalistica iscritta col n. 569 presso il Tribunale di Milano in data 14/10/2002. Pagina generata in 0.627 secondi. Sito ottimizzato per [Mozilla Firefox](#). Powered by [Kyron](#).

Programmazione.it v6.0 beta

Alessandro, hai già visitato l'[Oracle Database Center](#)? Se vuoi terminare la tua sessione, fai il [logout](#).

Pubblicità: [Canone Telecom? No grazie. TELE2: urbane e interurbane da 8,90 €/mese.](#)



Scopri la differenza

Microsoft

Pubblicità: [Canone Telecom? No, grazie. TELE2: urbane e interurbane illimitate a partire da 8,90 €/mese.](#)

Intervista al padre del Web, Tim Berners-Lee

Scritto da [Alessandro Carichini](#) il 08-01-2007 ore 13:08

Sul sito [Simple Talk](#), [Richard Morris](#) commenta un'intervista di [Tim Berners-Lee](#), rilasciata ad una trasmissione radiofonica della [BBC](#) e disponibile in [streaming](#) in lingua originale. Il tema è abbastanza scontato, ovvero **il presente e il futuro del Web** visto dagli occhi del suo creatore.

Per [Tim Berners-Lee](#) il presente non è dei migliori soprattutto per la china che sta prendendo [Internet](#). E' quella che lui definisce la **legge delle conseguenze inattese**, poiché il Web sta diventando non solo il mezzo preferito da imbroglioni, bugiardi e pornografi, ma anche la **cassa** di risonanza per l'emergere di un certo spirito antidemocratico. A queste affermazioni c'è chi minimizza sostenendo che il Web non possa essere usato solamente dai *bravi ragazzi*, c'è chi si fa una grassa risata, ma c'è anche chi, facendo le dovute premesse, sostiene che l'idea di base di voler proteggere la privacy degli utenti sia un fine da perseguire.

Ed è in quest'ottica che il [World Wide Web Consortium \(W3C\)](#) – fondato dallo stesso [Berners-Lee](#) - ha sviluppato un progetto chiamato [P3P \(Platform for Privacy Preference Project\)](#), che permette agli utenti di sapere in che modo i propri dati verranno utilizzati dal [Web Server](#) per decidere se fornirli o meno. Si tratta, in pratica, di uno standard tecnico per la definizione formale delle politiche di gestione dei dati personali.

Lo stesso [Berners-Lee](#) non è così ingenuo da bollare l'attuale Web come un male assoluto, tiene solo a precisare che occorre filtrare il bene dal male. Prendiamo l'esempio dei **Weblogs** che considera l'invenzione, che più si allinea con la sua idea originale di scrittura e lettura sul Web.

"Per anni ho cercato di far capire che, per la maggior parte delle persone, il Web non fosse uno spazio creativo. Editare pagine Web era difficile e complicato per le persone. Quello che è successo con gli spazi Web editabili come i blog e il wiki ha reso il tutto molto più semplice. Sono felicissimo di constatare che quando si scrive un blog non occorra scrivere ipertesti complicati, ma solo testo".

Purtroppo la parte negativa dei blog è che possono dare una credibilità assoluta nei confronti di persone, che non si conoscono affatto. La fiducia non si può basare su di un sistema così semplice, è necessario creare un sistema tecnologicamente più complesso. Ma la missione di **ricostruire la fiducia nel Web** per [Sir Timothy](#) non è affatto facile, e l'idea di mettere insieme l'[Università di Southampton](#) con il [Massachusetts Institute of Technology \(MIT\)](#) attraverso il [Web Science Research Initiative \(WSRI\)](#) nasce dalla volontà di sfruttare il Web per costruire un valido e serio strumento di ricerca scientifica. Per citare le parole del suo creatore: *"Its great potential as a powerful tool for humanity"*.

L'ultima sfida riguarderà l'evoluzione stessa di Internet attraverso il [Web semantico](#), che permetterà ai dati di essere condivisi e riutilizzati trasversalmente da tutte le applicazioni. Questo ci introduce al concetto di *tag descrittivo* o [RDF \(Resource Description Framework\)](#), dove le informazioni presenti al suo interno potranno essere usate in modo efficiente da altre applicazioni formando così nuove associazioni tra pezzi di informazione per creare, di fatto, un **database globale distribuito**.

Come sostiene [Richard Morris](#), [Thimoty Berners-Lee](#) è precisamente il tipo di eroe di cui noi tutti abbiamo più bisogno e se l'intervista non vi ha dato questa sensazione allora vi consiglio di leggersi il bellissimo libro ["L'architettura del nuovo Web"](#), che tratta approfonditamente la storia del World Wide Web per voce del suo stesso creatore.



[Commenta](#)

Precedente: [REALbasic 2007r1](#)

Successiva: [Le insidie del protocollo FTP: 5 soluzioni per proteggersi \(4/4\)](#)

Pubblicità: [ADSL GRATIS? Cogli l'occasione! Ancora per pochi giorni le offerte TELE2 per navigare gratis.](#)

Copyright Programmazione.it™ 1999-2006. Alcuni diritti riservati. Testata giornalistica iscritta col n. 569 presso il Tribunale di Milano in data 14/10/2002. Pagina generata in 0.581 secondi. Sito ottimizzato per [Mozilla Firefox](#). Powered by [Kyron](#).

Programmazione.it v6.0 beta

Ciao, per farti riconoscere devi fare il [login](#). Non ti sei ancora iscritto? Che aspetti, [registrati adesso!](#)

Publicità: [ADSL GRATIS? TELE2! Abbonati entro il 18/1, non paghi fino al 18/3.](#)



Cerca negli Articoli

Cerca nei Libri

Cerca nel Forum

0 cent
al minuto
15 cent
alla risposta

URBANE DA 8,90 € AL MESE

**Piccolo prezzo,
grande offerta.**

TELE2
Perché pagare di più?

Publicità: [ADSL GRATIS? COGLI al volo l'offerta di TELE2! Abbonati entro il 18/1, e non paghi fino al 18/3.](#)

Noie legali col file sharing? Compratevi un'isola!

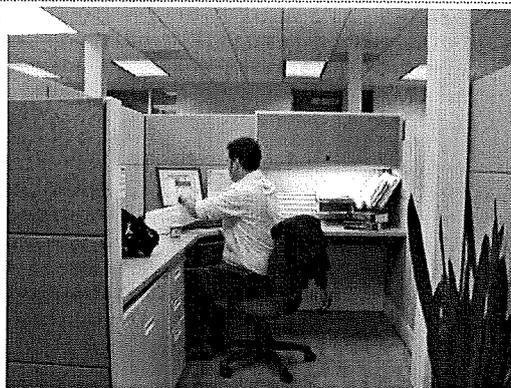
Scritto da [Alessandro Carichini](#) il 19-01-2007 ore 09:52

Il sito svedese di [file sharing](#) [The Pirate Bay](#) sta cercando di raccogliere delle donazioni per acquistare, non un nuovo [server](#), ma una vera e propria isola. Non siamo in presenza di romantici emuli di [Robinson Crusoe](#), che cercano un semplice appezzamento di terreno lontano dalle frenesie della società moderna, ma di un luogo con una sovranità indipendente dalle leggi internazionali sul copyright. In poche parole, una micronazione.

Si è costituito così un gruppo denominato **ACFI (Armed Coalition Forces of the Internet)**, che attraverso il sito [Buy Sealand](#) ha deciso di aprire una trattativa per acquistare il [Principato di Sealand](#), un vero e proprio territorio sovrano nato nel 1967. Non si tratta di una vera e propria isola ma di una **ex piattaforma militare di 550 mq in acque internazionali** a sud dell'Inghilterra con una connessione ad Internet ad alta velocità. Purtroppo però non viene riconosciuta da nessuna delle Nazioni Unite.

Il primo ostacolo da affrontare è l'ingente cifra [richiesta](#) dai suoi [sovrani](#) che ammonta a ben **765 milioni di euro** pari a 504 milioni di sterline. In caso di mancato accordo il piano B prevede l'acquisto di un vero isolotto da dichiarare indipendente un po' come fece a suo tempo [Paddy Roy Bates](#) quando si proclamò sovrano di [Sealand](#). A tal proposito consiglieri un [wiki](#) su come acquistare un'isola privata.

C'è anche chi ha preso tutta questa vicenda con molta ironia notando come la posizione geografica di [Sealand](#) fosse una sorta di *ombelico del mondo* per tutti gli [sviluppatori Debian](#). Da ciò al fantasticare di vincere una notevole cifra alla lotteria per investirla nella creazione di una vera nazione libera chiamata [Debian Land](#) il passo è breve.



[Commenta](#)

Precedente: [Exchange Server 2007: cosa ci aspetta 2/5](#)

Successiva: [HD DVD o Blu-ray? L'industria del porno ha già deciso.](#)

Publicità: [Canone Telecom? No, grazie. TELE2: urbane e interurbane illimitate a partire da 8,90 €/mese.](#)

Copyright Programmazione.it™ 1999-2006. Alcuni diritti riservati. Testata giornalistica iscritta col n. 569 presso il Tribunale di Milano in data 14/10/2002. Pagina generata in 0.547 secondi. Sito ottimizzato per [Mozilla Firefox](#). Powered by [Kyron](#).

Programmazione.it v6.0 beta

Ciao, per farti riconoscere devi fare il [login](#). Non ti sei ancora iscritto? Che aspetti, [registrati adesso!](#)

Publicità: [Canone Telecom? No, grazie. TELE2: urbane e interurbane illimitate a partire da 8,90 €/mese.](#)



Cerca negli Articoli

Cerca nei Libri

Cerca nel Forum

Publicità: [Canone Telecom? No, grazie. TELE2: urbane e interurbane illimitate a partire da 8,90 €/mese.](#)

I pigri della ricerca e la signorina Dewey

Scritto da [Alessandro Carichini](#) il 06-02-2007 ore 17:19

Quando dobbiamo fare una ricerca sul [Web](#), può essere seccante il dover tenere aperti diversi motori. Anche se alla resa dei conti la maggior parte di noi usa solamente [Google](#), quale può essere un metodo rapido per controllare parallelamente una ricerca su più motori? La risposta è un [search aggregator](#), ma vediamo i più interessanti.

Dietro ad nome da "*codice fiscale*" come [SRCHR](#) si racchiudono siti come [Google](#), [Flickr](#), [YouTube](#), [MySpace](#), [eBay](#), e [Eventful](#) - che non sapevo neppure esistesse. La caratteristica più interessante è la possibilità di personalizzare la pagina di ricerca, attraverso un account, per includervi i motori di ricerca preferiti. L'aspetto negativo, per così dire, è l'utilizzo massiccio di [AJAX](#) che non aiuta quelle macchine un po' a corto di [memoria](#). Comunque la comodità è notevole: in un'unica pagina si vedono i primi dieci risultati per ciascun motore di ricerca. Sono previste inoltre più categorie di ricerca come: notizie, Web, video, immagini, shopping, sociale, blog, lavori ed eventi.

Un altro sito simile a [SRCHR](#), ma con un'interfaccia più leggera e un range di [siti Web](#) inferiore, è [Dogpile Web Search](#), che oltre all'onnipresente [Google](#), offre solo [Yahoo!](#), [MSN Search](#) e [Ask.com](#). La lista di ciò che viene trovato è il risultato di un proprio sistema di *ranking* che non privilegia chi paga per avere i primi posti, ma chi è più popolare, ovvero chi ha avuto più accessi.

Come si dice in alcuni casi "...e l'ultimo chiude la porta", [CrossEngine](#) se li può tranquillamente lasciare alle spalle gli altri motori. Diciamo che miscela abilmente la complessità di [SRCHR](#) con la fluidità di [Dogpile](#). Ha un notevole numero di motori da cui attingere e la provenienza dei risultati della ricerca è palese.

Se però l'avvenenza femminile ha priorità su tutto, anche sulla qualità dei risultati, allora vi consiglio il nuovo sito basato sul [Windows Live](#) di Microsoft, denominato [Ms Dewey](#), nel quale l'attrice americana [Janina Gavankar](#), mossa da oltre 600 animazioni in [Flash](#), prende sul personale ogni domanda che le rivolgete - alterandosi pure in alcuni casi - per poi lasciare spazio ai risultati veri e propri della ricerca. A volte anche l'occhio vuole la sua parte!



[Commenta](#)

Precedente: [Un tutorial per la pubblicazione di un proprio canale PEAR](#)

Publicità: [Cogli l'occasione: TELE2 ADSL 2 MESI GRATIS. Abbonati entro l'8/2.](#)

Copyright Programmazione.it™ 1999-2006. Alcuni diritti riservati. Testata giornalistica iscritta col n. 569 presso il Tribunale di Milano in data 14/10/2002. Pagina generata in 0.831 secondi. Sito ottimizzato per [Mozilla Firefox](#). Powered by [Kyron](#).

Programmazione.it v6.0 beta

Ciao, per farti riconoscere devi fare il [login](#). Non ti sei ancora iscritto? Che aspetti, [registrati adesso!](#)



Cerca negli Articoli

Cerca nei Libri

Cerca nel Forum



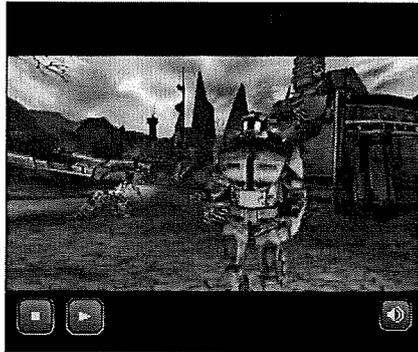
Un find in stile UNIX per MySQL

Scritto da [Alessandro Carichini](#) il 16-05-2007 ore 12:04

Dopo aver parlato dei diversi strumenti presenti nel [MySQL Toolkit](#) ecco che ne spunta uno nuovo che risponde alla domanda: "Esiste anche per MySQL un corrispettivo del find di UNIX?". Se vi siete abituati ad usare il find per qualcosa di più serio del semplice trova e stampa, allora ne conoscete la reale potenza.

Saprete infatti che si tratta di una sorta di ambiente di programmazione miniaturizzato per trovare e manipolare file e directory. Così il prolifico [Baron Schwartz](#) ne ha tratto l'ispirazione per realizzare il [MySQL Find](#), un'utilità scritta in Perl — come del resto tutte le altre del Toolkit — per risolvere, attraverso la linea di comando, quello che molto spesso deve essere demandato a qualche script *ad hoc* o a qualche *megainterfaccia* grafica dalle mille opzioni.

Tralasciandone l'aspetto più comune come la ricerca di tabelle e indici, il [MySQL Find](#) dà il meglio di sé nella gestione delle cosiddette [tabelle temporanee](#) intese sia come *TEMPORARY*, che come *scratch*: le prime create attraverso l'opzione del comando SQL *CREATE TABLE*, le seconde come vere e proprie tabelle di appoggio per operazioni di analisi e controllo a lungo termine. Qui di seguito riporto gli esempi più significativi:



Per eliminare quelle scratch table create da processi già terminati:

```
mysql-find --pid 'D_(d+)$' scratch --exec_plus "DROP TABLE %s"
```

Per eliminare delle scratch table per delle analisi la cui ultima modifica risale ad un mese fa:

```
mysql-find --mtime +30 analyst_scratch --exec_plus "DROP TABLE %s"
```

Per salvare le dimensioni della tabella ed il numero di record per l'analisi e controllo a lungo termine:

```
mysql-find --noquote --exec "INSERT INTO stat.tblsize(db, tbl, idxlen, datalen, rowcount) VALUES('%D', '%N', %I, %d, %S)";
```

Per il momento [Baron Schwartz](#) ritiene di non dover apportare significative migliorie alla propria creatura, perché già include tutto ciò di cui aveva bisogno. Magari il *feedback* di qualche utilizzatore potrebbe fargli cambiare idea per rendere questo strumento ancora più flessibile.



[Commenta](#)

Precedente: [Versione aggiornata di Underground PHP and Oracle Manual](#)

Successiva: [DUGL, la DOS Ultimate Game Library](#)

Pubblicità: [Annunci auto Autodaytona](#), per [cercare auto nuove](#) e [vendere auto usate](#).

Copyright Programmazione.it™ 1999-2006. Alcuni diritti riservati. Testata giornalistica iscritta col n. 569 presso il Tribunale di Milano in data 14/10/2002. Pagina generata in 0.776 secondi. Sito ottimizzato per [Mozilla Firefox](#). Powered by [Kyron](#).

Programmazione.it v6.0 beta

Ciao, per farti riconoscere devi fare il [login](#). Non ti sei ancora iscritto? Che aspetti, [registrati adesso!](#)



Cerca negli Articoli

Cerca nei Libri

Cerca nel Forum

difendere il sistema è ancora più facile.



Come sviluppare un'applicazione video con meno di 1000 righe di codice

Scritto da [Alessandro Carichini](#) il 29-05-2007 ore 12:10

La maggior parte degli utenti avanzati di [Linux](#) conosce [FFmpeg](#) come un potentissimo toolkit a riga di comando per la conversione dei formati video tanto che per essere utilizzato anche dai meno esperti è stata creata un'interfaccia grafica semplice ed intuitiva come [SUPER](#).

Per la verità [FFmpeg](#) è anche una libreria che assieme al suo codec [libavcodec](#) può essere usata per costruire applicazioni, che consentano di registrare, convertire e fare streaming audio/video in modo trasparente. Fino a poco tempo fa l'unica documentazione degna di nota era [Using libavformat and libavcodec](#) di [Martin Bohme](#), ed invece ora, grazie al sito [ranger.com](#) abbiamo un ampio [Tutorial](#) per creare attraverso il [Simple Directmedia Layer \(SDL\)](#) ed il linguaggio C un player video completo con poco meno di 1000 righe di codice.

Il [Tutorial](#) si divide in otto capitoli (grosso modo una ventina di pagine) ognuno dei quali ci introduce a nuovi concetti o funzionalità come la [cattura dello schermo](#), lo [spawning thread](#), la [sincronizzazione audio/video](#) e la [ricerca rapida](#) (*fast forward* e *rewind*) spiegandoci come implementarli attraverso esempi e sorgenti in C. [FFmpeg](#) è sviluppato sotto Linux, ma può essere compilato anche sotto altri sistemi operativi come Windows, come ampiamente documentato nel [wiki](#).

fermare gli hackers.
ancora più facile.



[Commenta](#)

Precedente: [Ruby per applicazioni web](#)

Successiva: [Integrare Linux in Active Directory \(1/2\)](#)

Pubblicità: [Annunci auto Autodaytona](#), per cercare auto nuove e vendere auto usate.

Copyright Programmazione.it™ 1999-2006. Alcuni diritti riservati. Testata giornalistica iscritta col n. 569 presso il Tribunale di Milano in data 14/10/2002. Pagina generata in 1.874 secondi. Sito ottimizzato per [Mozilla Firefox](#). Powered by [Kyron](#).

Programmazione.it v6.0 beta

Ciao **Alessandro**, benvenuto. (Se non sei Alessandro Carichini o vuoi terminare la tua sessione, fai il [logout](#).)



TONY: Pronto?

TOM: Pronto.

rolover per continuare

Microsoft
Visual Studio

Per saperne di più clicca qui

Uno script per verificare le novità sui siti Web

Scritto da [Alessandro Carichini](#) il 08-06-2007 ore 12:37

I feed RSS e le newsletter sono ottimi strumenti, che ci informano immediatamente sulle novità pubblicate da quei siti Web che maggiormente ci interessano, sfortunatamente però non tutti li adottano per cui è necessario visitarli costantemente. Per risolvere la questione abbiamo due alternative: affidarci ad applicazioni open source e *cross platform* come [WebMonX](#) e [Webalizer](#), oppure sfruttare la potenza del [KISS \(Keep It Simple, Stupid\)](#) ed automatizzare questa noiosa operazione di verifica quotidiana con un piccolo script in [Bash](#).

Questo semplicissimo script è stato recentemente pubblicato su [Linux.com](#) per opera di [Leslie P. Polzer](#), *project manager* di [GNU Parted](#) e [GNU Fdisk](#). L'idea è quella di attingere i nomi dei siti da visitare da un comunissimo file di testo per verificare di volta in volta se ci sono state delle modifiche.

Per ogni sito viene creato un file il cui nome viene generato, grazie a [md5sum](#), dall'[Hash](#) del proprio URL e al suo interno viene inserito il *dump* della pagina iniziale utilizzando un Web browser testuale come [w3m](#). Mettendo una schedulazione giornaliera dello script possiamo ricevere delle notifiche via e-mail nel caso venissero riscontrate delle differenze.

```
#!/bin/sh
# webtrack.sh

# Mail dove inviare le notifiche
RECIP=user@localhost
# Richiamo del browser di testo
DUMPCMD="w3m -dump"
# File contenente i siti da visitare
WWWLIST="list.txt"

for url in $(cat $WWWLIST); do
md5=$(echo "$url" | md5sum | cut -d -f 1)
touch $md5.txt
$DUMPCMD "$url" > tmp.txt
if diff $md5.txt tmp.txt >/dev/null; then
: #echo nessun cambiamento
else
: #echo "cambiamenti: "
diff -Napu $md5.txt tmp.txt > diff.txt
mv tmp.txt $md5.txt
BODYMAIL="diff ha rilevato $(wc -l diff.txt | cut -d -f 1) righe $(cat diff.txt)"
echo "$BODYMAIL" | mail -s "Il sito $url è stato aggiornato" "$RECIP"
fi
done
```

Lo script può essere ulteriormente ampliato ed integrato con un semplice *data-entry* per l'inserimento dei siti da verificare, oppure trasformando la notifica via e-mail con un [feed RSS](#) autoprodotta. Le possibilità da questo punto di vista certamente non mancano.



[Commenta](#)

Precedente: [Un ponte tra le virtualizzazioni di Xen e KVM](#)

Successiva: [Cansdale sviluppa un add-in per Visual studio, Microsoft lo minaccia](#)

Pubblicità: [Annunci auto Autodaytona](#), per cercare [auto nuove](#) e vendere [auto usate](#).

Intervento di **Antonio Orlando** a.k.a. [insert_nick](#) del 16-06-2007 ore 17:25, Bari (BA)



Segnalo a tal proposito l'ottimo servizio web gratuito:

Programmazione.it v6.0 beta

Ciao **Alessandro**, benvenuto. (Se non sei Alessandro Carichini o vuoi terminare la tua sessione, fai il [logout](#).)





Per te, gratis, Visual Studio 2008!

[Clicca qui per saperne di più](#)

Microsoft
Visual Studio 2008

Colture artificiali di neuroni per memorizzare informazioni

Scritto da [Alessandro Carichini](#) il 12-06-2007 ore 09:59

Sembra quasi il *plot* di un romanzo di [William Gibson](#), un ipotetico seguito di [Johnny Mnemonic](#), ed invece è pura realtà. Infatti due ricercatori dell'[Università di Tel Aviv](#), [Itay Baruchi](#) ed [Eshel Ben-Jacob](#), hanno scoperto la possibilità di memorizzare delle informazioni all'interno di una coltura artificiale di neuroni attraverso la continua stimolazione chimica.

Questa nuova tecnica ha rivelato la notevole capacità di apprendimento di una rete di neuroni che, sotto il fuoco di continue stimolazioni, genera spontaneamente dei nuovi legami tra le cellule: una sorta di bit di informazione. Questo può essere il primo passo verso l'avveniristica creazione di un chip [cyborg](#) che combini circuiti elettronici con cellule umane, molto utile per monitorare quei sistemi biologici come cervello e sangue, che rispondono della medesima chimica.

Per maggiori informazioni rimando all'articolo originale scritto dai due ricercatori intitolato: [Towards neuro-memory-chip: Imprinting multiple memories in cultured neural networks](#), in cui vengono spiegate dettagliatamente le fasi dell'esperimento.



[Commenta](#)

Precedente: [Cinque script per semplificarci la vita con vim](#)

Successiva: [ITIL, pronta la versione 3](#)

Pubblicità: [Annunci auto Autodaytona](#), per cercare [auto nuove](#) e vendere [auto usate](#).

Intervento di **Fabrizio Furnari** a.k.a. [gwfwg](#) del 12-06-2007 ore 11:38, Anguillara sabazia (RM)



Nobile



(74 interventi)

Iscritto il 17-10-2006

Grande Gibson!



[Rispondi](#)

Programmazione.it v6.0 beta

Ciao **Alessandro**, benvenuto. (Se non sei Alessandro Carichini o vuoi terminare la tua sessione, fai il [logout](#).)



TONY: Pronto?

TOM: Pronto.

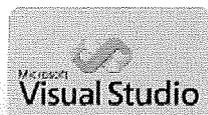


Microsoft
Visual Studio

Per saperne di più [clicca qui](#)

Microsoft

Con le applicazioni dinamiche per Vista.
Vinci tutte le sfide con i corsi Learn&Get.



Per saperne di più.

[Clicca qui](#)

Vent'anni di GNU C Compiler!

Scritto da [Alessandro Carichini](#) il 14-06-2007 ore 11:19

Erano i primi di giugno del 1987 quando [Richard M. Stallman](#) annunciò la versione 1.0 del proprio [GNU C Compiler](#) appena due anni dopo la pubblicazione dell'emblematico [Manifesto GNU](#).

Nello stesso anno Microsoft esce con Windows 2.0 — anche se è ufficialmente impegnata con IBM allo sviluppo di [OS/2 Warp](#); Apple presenta il [Macintosh II](#) ed [Andrew S. Tanenbaum](#), emerito professore di Informatica ad Amsterdam, crea [Minix](#), il primo UNIX che poteva girare sotto architettura Intel x86, da cui poi [Linus Torvalds](#) prenderà spunto per il suo Linux, che uscirà ben cinque anni più tardi. E questo anche grazie all'apporto del [GCC](#).

La nuova creatura di [Stallman](#) influenzò la vita di parecchi programmatori, tra cui [Micheal Tiemann](#) — fondatore di [Cygnus Solutions](#) ed ora vicepresidente di [Red Hat](#) — che a quel tempo, poco più che ventenne, sognava di poter scrivere quello che lui definiva "*The Great American Compiler*". Nasce così il GNU C++ Compiler ed una nuova ondata di sviluppatori avente come *missione* la diffusione del movimento Open Source. Suggestivo il "*come eravamo*" pubblicato sul proprio [blog](#).

Celebriamo quindi i vent'anni del [GNU C Compiler](#) appena giunto alla [release 4.2](#).



[Commenta](#)

Precedente: [Anatomia del kernel di Linux \(2/2\)](#)

Successiva: [Ophcrack LiveCD, la soluzione alle password perse](#)

Pubblicità: [Annunci auto Autodaytona](#), per cercare [auto nuove](#) e vendere [auto usate](#).

BE IT Mettiti alla prova. Vinci una Xbox con Halo 3. **Microsoft**

Programmazione.it v6.0 beta

Ciao, per farti riconoscere devi fare il [login](#). Non ti sei ancora iscritto? Che aspetti, [registrati adesso!](#)





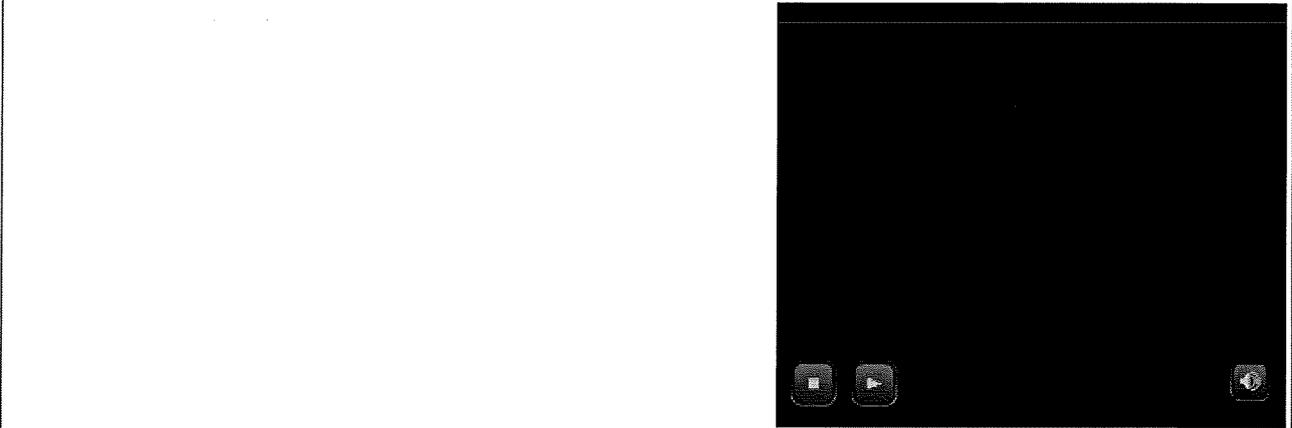


TONY: Pronto?
TOM: Pronto.

rolover per continuare

Per saperne di più clicca qui

Dal MIT un algoritmo per convertire i segnali cerebrali in azioni
Scritto da [Alessandro Carichini](#) il 08-10-2007 ore 12:52



Alcuni ricercatori del MIT ([Massachusetts Institute of Technology](#)) hanno sviluppato un nuovo algoritmo per convertire i segnali cerebrali in azioni, che dovrebbe aiutare la creazione di dispositivi protesici utili a quei pazienti che hanno subito una paralisi o l'amputazione degli arti.

La tecnica, che verrà descritta nell'articolo di copertina del numero di ottobre del [Journal of Neurophysiology](#), sembra unificare i tanti approcci presi dai diversi gruppi sperimentali, che creano i prototipi su quei dispositivi di protesi neurali in animali o umani. Fino ad ora infatti gli algoritmi utilizzati erano in funzione del metodo con cui si misurava l'attività del cervello. Secondo [Lakshminarayan "Ram" Srinivasan](#) — ricercatore presso il [Center for Nervous System Repair](#) del [Massachusetts General Hospital](#) di Boston — grazie a questa nuova tecnica non sarà più necessario reinventare un nuovo paradigma per ogni modalità o regione cerebrale.

Gli algoritmi formano il collegamento tra i segnali neurali che sono stati registrati e le intenzioni dell'utente, che sono state decodificate per guidare la protesi. Questi algoritmi rappresentano quindi un rapporto matematico tra le intenzioni della persona e la manifestazione neurale di quella intenzione se essa viene misurata da una [elettroencefalografia EEG](#) con elettrodi intracraniali o attraverso l'[optical imaging](#). Questi segnali possono provenire da un certo numero di regioni del cervello, comprese le strutture corticali o sottocorticali.

Con una comprensione quantitativa migliore di come il cervello controlla normalmente il movimento e i meccanismi della malattia si spera che questi dispositivi possano un giorno permettere un livello di destrezza simile al braccio meccanico del personaggio interpretato da Will Smith nel film "Io, Robot".


[Commenta](#)

Precedente: [V Convegno Net and System Security e Call for Papers](#)

Successiva: [Funzionalità non documentate in PGP Whole Disk Encryption](#)

Pubblicità: [Annunci auto Autodaytona](#), per cercare [auto nuove](#) e vendere [auto usate](#).

BE IT Mettiti alla prova. Vinci una Xbox con Halo 3. **Microsoft**

BE IT Mettiti alla prova. Vinci una Xbox con Halo 3. **Microsoft**

Programmazione.it v6.0 beta

Ciao, per farti riconoscere devi fare il [login](#). Non ti sei ancora iscritto? Che aspetti, [registrati adesso!](#)



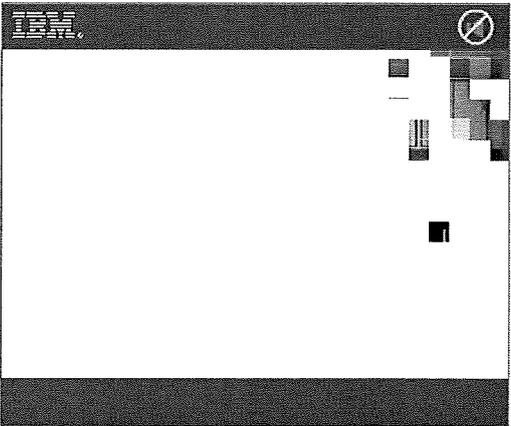
Microsoft

Acquista Visual Studio con MSDN.

Microsoft Visual Studio 2008

Intervista a Jim Starkey sul futuro dei RDBMS

Scritto da [Alessandro Carichini](#) il 12-11-2007 ore 09:26



[Jim Starkey](#) può essere considerato la massima autorità nel campo dei [database](#) relazionali: è stato il primo architetto di [InterBase](#), facilitando poi lo sviluppo di [Firebird](#) ed oggi, dopo l'acquisizione di [NetFrastructure](#) da parte di [MySQL AB](#), è sempre più impegnato nel progetto [Falcon](#), un motore transazionale che sarà l'alternativa a [InnoDB](#) dalla versione 6.0 di [MySQL](#).

Sul portale del [Dr.Dobb's](#) è stata pubblicata un'intervista nella quale [Starkey](#) parla di [open source](#) e di database relazionali. I vantaggi di sviluppare sotto il modello open source sono molteplici ed il più significativo è la capacità di spingere il programmatore a dare il meglio di sé: "Ad ogni linea di codice che scrivo immagino sempre le ipotetiche domande durante le sessioni tecniche delle conferenze [MySQL](#)".

In tema di database relazionali parla di come una delle prime pubblicazioni di [Edgar Frank Codd](#) sia stato il vero colpo di fulmine e di come questo modello si sia imposto nel corso degli anni. Tornando alla progettazione e di come riesca a bilanciare tematiche in perenne conflitto come la semplicità d'uso, la sicurezza e le performance, [Starkey](#) risponde che la facilità d'uso ha la priorità su tutto.

Riguardo alla sicurezza il discorso è più complesso perché riguarda nel profondo il modello SQL progettato per sistemi in *time-sharing*, dove tutti condividevano i database attraverso dei terminali "stupidi". Per questo è giunto il momento che si prenda atto che il controllo degli accessi e la visibilità non possono più essere gestiti al solo livello di tabella, ma devono esserlo anche a livello di record visto che l'accesso al [server](#) non rappresenta più una connessione ad un client fisico.

Ad incidere pesantemente sulle performance ci sono i tipi di dati e l'esempio più classico è l'adozione di campi a lunghezza variabile a dispetto di quelli a lunghezza fissa. [Falcon](#) ad esempio memorizza i dati per valore attuale piuttosto che per tipo dichiarato. Anche per questo [Starkey](#) si augura per i database quello che già avviene per i linguaggi moderni, ovvero il poter gestire delle stringhe non dimensionate. La forza del modello relazionale è a prescindere dai tipi di dati che implementa, della topologia dell'applicazione e dalle interfacce di rete.


[Commenta](#)

Precedente: [the Essential Blender](#)

Successiva: [Creare documenti XPS in .NET](#)

Pubblicità: [Annunci auto Autodaytona](#), per cercare [auto nuove](#) e vendere [auto usate](#).

BE IT Mettiti alla prova. Vinci una Xbox con Halo 3. **Microsoft**