

Programmazione.it v6.2

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





Orientarsi tra classi e metodi di Java

Scritto da [Paolo Raviola](#) il 24-06-2009 ore 08:19



Alcuni ricercatori della [School of Computer Science](#) alla Carnegie Mellon University hanno messo a punto due strumenti per venire in aiuto degli sviluppatori Java, che devono orientarsi tra migliaia di metodi e classi.

I due tool, [Jadeite](#) (Java Documentation with Extra Information Tacked-on for Emphasis) e [Apatite](#) (Associative Perusal of APIs That Identifies Targets Easily), si basano su tecniche incentrate sull'esperienza dei programmatori, in modo da ridurre significativamente il tempo dedicato ai tentativi per trovare le API giuste, necessarie a portare a termine il lavoro desiderato. [Brad A. Myers](#), professore di *human-computer interaction*, fa notare che anche il più esperto degli sviluppatori non può avere familiarità con circa 35.000 metodi e 4.100 classi, che si vanno facendo sempre più numerose a ogni nuova release.

[Jadeite](#) **augmenta l'usabilità migliorando l'attuale documentazione Javadoc** in un modo già in uso in parecchi siti. Le dimensioni dei caratteri corrispondenti alle funzioni più usate, come *PrintWriter*, sono più grandi e più in evidenza rispetto a quelle delle funzioni meno utilizzate, come *PrintEvent*.

Ma l'aspetto forse più interessante è che si possono inserire dei rimandi a metodi che un utente si aspetterebbe di trovare in una certa classe. Per esempio, le classi *Message* e *MimeMessage* non includono un metodo per inviare un'e-mail. [Jadeite](#) consente di mettere dei segnaposto (*placeholder*), che guidano il programmatore alle funzioni volute, o che spiegano perché non esiste un'API, che corrisponda alla funzionalità desiderata.

Si trovano inoltre esempi del codice più comunemente usato per creare alcuni oggetti, che potrebbero risultare ostici ai neofiti. Per esempio, creare un SSL socket, come nell'esempio sotto riportato:

```
1. SSLSocketFactory factory = ...;
2. String host = ...;
3. int port = ...;
4. SSLSocket socket = (SSLSocket)factory.createSocket(host, port);
```

[Apatite](#) **permette invece ai programmatori di esaminare le API per associazione**, evidenziando quali pacchetti, classi e metodi tendono a essere usati insieme. Viene inoltre usata una statistica della popolarità, per fornire delle vedute degli item più usati. È stato utilizzato un metodo iterativo per adattare il tool alle esigenze dei programmatori.

[Myers](#) dice che entrambi sono strumenti di ricerca, ma comunque sono disponibili pubblicamente, nella speranza che gli utenti contribuiscano a migliorarli. Essi fanno parte del [Natural Programming Project](#), un'iniziativa all'interno dello [Human-Computer](#)

[Interaction Institute](#), per rendere più facile la programmazione.



[Commenta](#)

Precedente: [Creare oggetti SCORM con pochi clic](#)

Successiva: [Browserlabs e altri, testare un sito su più browser \(3/4\)](#)

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

[Copyright](#) Programmazione.it™ 1999-2009. Alcuni diritti riservati. Testata giornalistica iscritta col n. 569 presso il Tribunale di Milano in data 14/10/2002. Pagina generata in 0.908 secondi. Sito ottimizzato per [Mozilla Firefox](#). Powered by [Kyron](#).