

Introduzione al profiling di applicazioni

Ludovico Pavesi

2020-10-24

Chi sono io

Un **umile programmatore** C++ e talvolta LISP in una grande azienda.

Per qualche motivo scrivo anche codice PHP e Python e amministro server e tante altre belle cose per WEEE Open (weeeopen.polito.it).

Cos'è il profiling

Tecnica di **analisi** del software finalizzata all'**ottimizzazione**.

Può riguardare:

- Tempo impiegato (da ogni istruzione, funzione, blocco di codice, etc...)
- Istruzioni eseguite (da ogni funzione, blocco di codice, etc...)
- Memoria occupata
- Cache hit/miss

Ottimizzazione: meno istruzioni eseguite, meno tempo impiegato, meno memoria occupata, etc...

Vantaggi per:

- Utenti: software più reattivo, requisiti di sistema più bassi
- Server: meno server necessari o meno potenti, ogni server può gestire più utenti
- Ambiente: di solito meno istruzioni eseguite, meno elettricità consumata

Premature optimization is the root of all evil — Donald E. Knuth

Come si fa

Con un profiler.

Quale utilizzare **dipende** dal linguaggio, integrazione con IDE e altri strumenti, etc...

Vedremo:

- Profiler di tempo/istruzioni: Callgrind, VisualVM
- Visualizzatori: KCacheGrind, Qt Creator, VisualVM

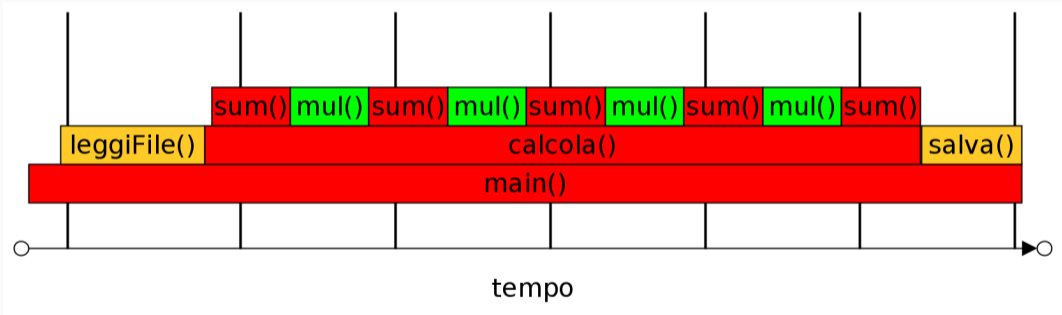
Concetti generali

Due tipi di profiler:

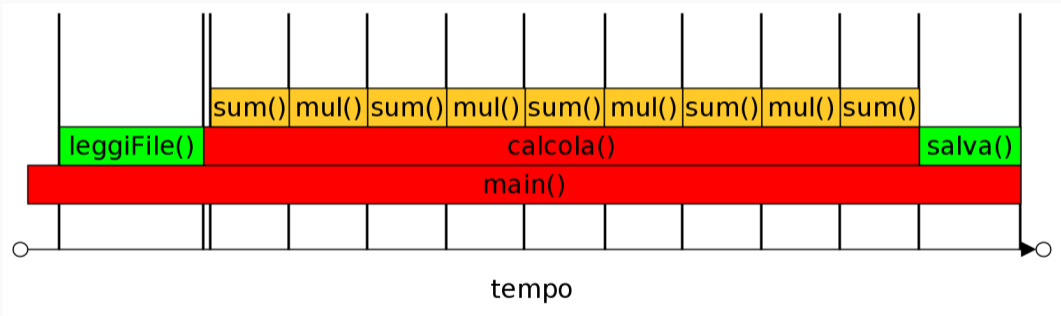
- **Sampling**: misure a campione ogni tot
- **Instrumented**: dati precisi forniti da VM, interprete, debugger, software stesso, ...

	Precisione	Velocità
Sampling	Bassa	Alta
Instrumented	Alta	Bassa

Sampling



Instrumented



Passiamo al codice!

Fine

Grazie per l'attenzione

Domande?

Codice sorgente: github.com/lvps/talk-profiler-linuxday-2020