



Orientarsi nel mondo open source

Alessandra Cau
<http://agile.diee.unica.it/cau.html>



Introduzione

- Le soluzioni open source (OS) si stanno diffondendo per almeno due motivi
 - Costo: Il software open source è solitamente distribuito gratuitamente
 - Qualità: OS vuol dire che il codice sorgente risulta accessibile, la trasparenza del codice permette di raggiungere alti livelli di qualità.



Introduzione

- Risulta difficile però scegliere quale prodotto OS usare, in quanto il numero di prodotti OS sta crescendo a dismisura
- Come reperire i prodotti OS?
 - siti web dei prodotti
 - Repository Open Source



Repository

- SourceForge, (<http://sourceforge.net/>), ospita oltre 100,000 prodotti OS
- Freshmeat, (<http://freshmeat.net/>)
- CodeHaus, (<http://codehaus.org>)
- Tigris, (<http://www.tigris.org/>)
- Java.net, (<http://java.net/>)



Orientarsi nei Repository

- Esempio: prodotto CMS (Content Management System)
- Come scegliere tra le varie alternative?
- Su SourceForge non esiste una categoria specifica per i CMS
- Se si stanno cercando CMS scritti in Java
 - <http://java-source.net/>



Orientarsi nei Repository

- <http://java-source.net/open-source/>
 - categoria: Content Management Systems (CMS) in Java
- Quale di questi prodotti non fallirà in breve tempo?
- Quale di questi prodotti è di qualità maggiore?
- Quale di questi prodotti è il più usato?



Obiettivo

- In questa presentazione discuteremo delle metriche che devono essere considerate per scegliere un progetto open source.



Successo

- Nel caso di progetti open source riuscire a misurare il successo è estremamente importante perché diventa un modo per valutare se un prodotto sarà affidabile nel lungo termine e per evitare che un manager IT che sceglie un certo prodotto open source faccia un investimento fallimentare.



Successo

- Caratteristiche proprie del processo che possono influenzare il successo
- Qualità del codice
- Adozione: penetrazione del prodotto nel mercato



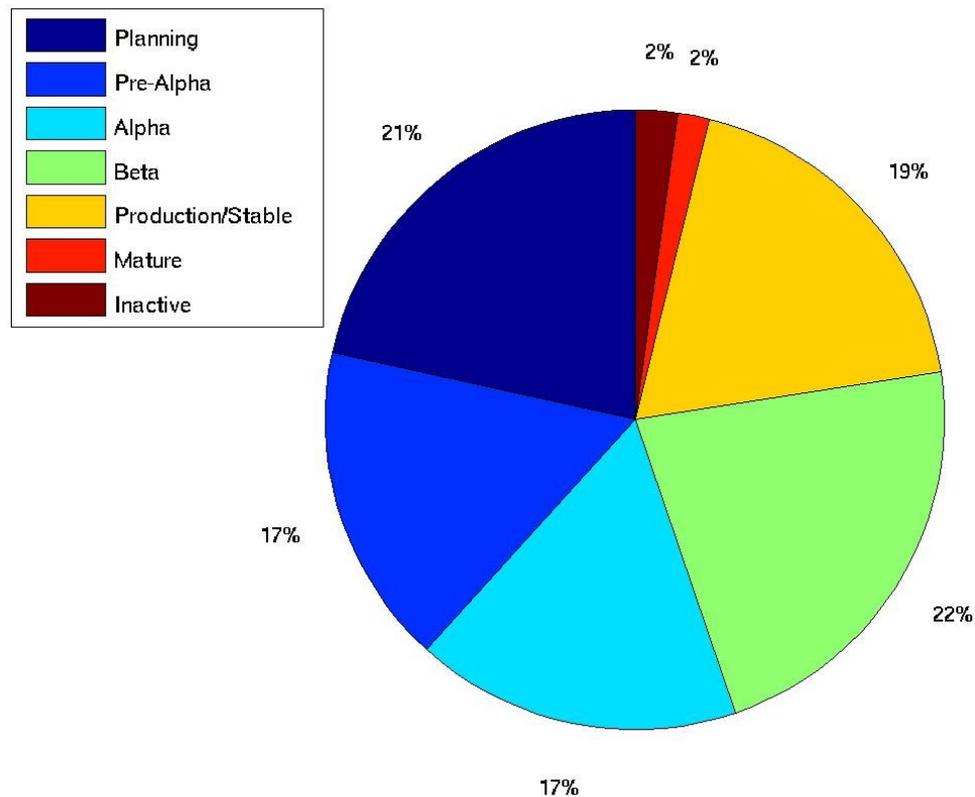
Caratteristiche del Processo

- Utilizzo di un sistema di Version Control
- Età: la maggior parte dei prodotti open source fallisce nel primo anno di vita
- Numero di sviluppatori: Abilità di un progetto di attrarre sviluppatori
- Utilizzo di Mailing list
- Status: è importante in quanto solo una piccola parte dei progetti OS raggiunge alti livelli di maturità



Caratteristiche del Processo

- Stato dei progetti ospitati in Source Forge





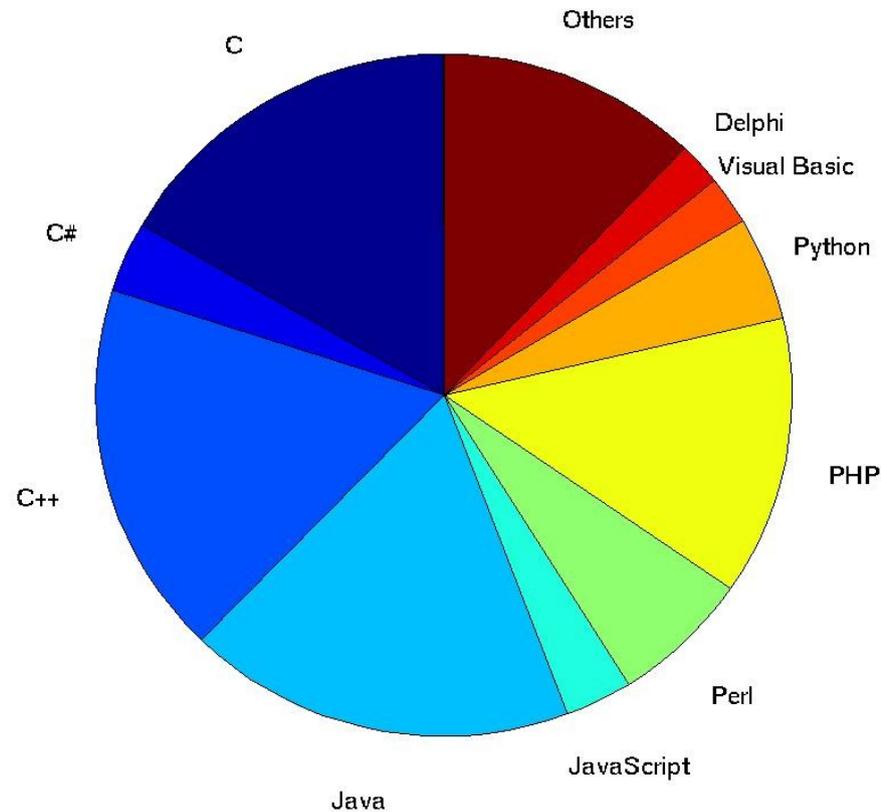
Qualità del Codice

- La qualità di un prodotto è legata al successo in quanto la mancanza di qualità di un prodotto può avere disastrose conseguenze
- Alta qualità di un prodotto riduce il rischio associato all'uso del prodotto
- La misura + comune di qualità del codice è il
 - Numero di difetti per Kloc
- Negli ultimi anni i linguaggi OO stanno acquistando molto successo e nuove metriche che valutano le proprietà OO stanno diventando molto popolari.
 - Metriche CK



Qualità del Codice

- Più del 40% dei prodotti ospitati da SF è OO





WMC

- Numero pesato di metodi di una classe
- Peso unitario \Rightarrow WMC = Numero di metodi della classe
- Raramente si usa come peso la complessità ciclomatica (metrica tradizionale)
- Questa metrica non deve essere troppo alta. Classi con troppi metodi sono difficili da leggere e da modificare.



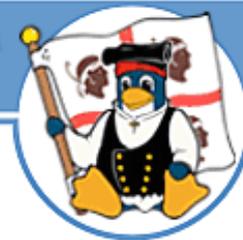
RFC

- E' il numero di metodi che potenzialmente possono essere invocati in risposta ad un messaggio inviato ad un oggetto della classe.
- RFC = somma del numero di metodi della classe + numero di metodi chiamati (di classi diverse e senza duplicati).
- RFC = outlinks di un grafo diretto e pesato



CBO

- CBO misura il numero di classi diverse con cui una data classe è accoppiata.
- Una classe A è accoppiata con una classe B se usa almeno un metodo di B o accede ad almeno una variabile di B.
- Valori alti di CBO comportano maggiore propensione della classe ai difetti e alla scarsa manutenibilità.
- CBO misura gli outlinks di un grafo diretto e non pesato



LCOM

- E' una misura della mancanza di coesione di una classe
- La coesione di una classe deve essere alta, la mancanza di coesione deve essere bassa.
- Una classe è coesa quando i suoi metodi sono coesi: due metodi sono coesi se usano almeno una stessa variabile (di istanza o di classe).
- LCOM è la differenza tra il numero di coppie di metodi che non usano nessuna variabile condivisa e il numero di coppie che usano almeno una variabile in comune.



DIT, NOC

- DIT: distanza di una classe dalla radice dell'albero di ereditarietà
- NOC: numero di sottoclassi di primo livello
- NOC è il numero di inlinks di un grafo orientato, non pesato, in cui i link sono relazioni di tipo “extends”.



Adozione

- Per misurare l'utilizzo di un software proprietario è sufficiente contare il numero di licenze vendute
- Per software open source è difficile misurare l'attuale uso di un prodotto, per quantificare il livello di adozione di un prodotto è possibile valutare il potenziale uso di esso (popolarità)



Adozione

- Metriche di popolarità raccolte da FLOSS repository (<http://sourceforge.net/>)
 - Number of Downloads
 - Number of Pages Views
 - Number of Hits
- Queste ultime due metriche per la loro stessa natura risultano altamente correlate
- Sono facilmente calcolabili, ma risultano dipendenti dal specifico repository
- Trovare delle metriche indipendenti dal repository



Adozione

- Popolarità su Web è considerata una buona stima del successo del progetto
- Metrica: Somma del numero di backlinks all'home page del progetto
- Backlinks sono definiti come:
 - Se una pagina A ha un link verso la pagina B, definiamo un backlink come il link che dalla pagina B porta alla pagina A. Dal punto di vista di un sito web, un backlink è un link ricevuto da un altro sito.



**Grazie per l'attenzione
Domande?**