

Дистрибутирани системи за управление на сорс код

Зорница Атанасова Костадинова, 4 курс, КН, фн: 80227,
Искрен Ивов Чернев, 4 курс, КН, фн: 80246

2 февруари 2011 г.

Цел на проекта

Да запознае читателя с историята и развитието на системите за управление на код (SCM). Ще бъдат сравнени централизираните и дистрибутираните системи както на високо ниво - предимства, недостатъци, сфери на приложение - така и на ниско ниво - обща архитектура, представяне на данните, използвани алгоритми и структури от данни. Ще бъде обърнато специално внимание на дистрибутираните системи за управление на код, тъй като те са по-нови като концепция и вече успешно заместват централизираните системи във все повече проекти, най-забележимо тези с отворен код, но също и в големи компании които по исторически причини използват централизирани системи (google, facebook).

Съдържание

1	Увод	3
2	История и развитие на SCM	4
2.1	Ръчно управление	4
2.2	Локално управление	4
2.3	Централизирано управление	4
2.4	Дистрибутирано управление	4
3	Сравнение на архитектурно ниво	4
3.1	Архитектура на централизираните системи	4
3.2	Архитектура на дистрибутираните системи	4
4	Сравнение на функционално ниво	4
4.1	Предимства на централизираните системи	4
4.2	Недостатъци на централизираните системи	4
4.3	Предимства на дистрибутираните системи	4
4.4	Недостатъци на дистрибутираните системи	4
5	Заклучение	4

1 Увод

Системите за управление на код (SCM) играят важна роля в процеса на разработване на софтуер. Те съхраняват историята на развитие на файловете като по този начин позволяват на потребителя да прегледа направените промени по различни критерии (времеви период, потребител направил промяната и др.). Правенето на промени по кодовата база също е благоприятствано от факта, че винаги може да се игнорират и състоянието на проекта да бъде върнато към по-старо и стабилно състояние. Историята на развитие на проекта може да бъде използвана и от хора интересуващи се от прогреса по проекта (мениджъри, клиенти), с цел създаване на план за по-нататъшно развитие, оценка за свършена работа и други.

Софтуерните проекти обикновено се развиват в няколко направления:

- едно или няколко стабилни направления - използват се от обикновения потребител;
- направление за тестване (beta версия) - използват се от по-напреднали потребители, които искат да получат нововъведенията колкото се може по-скоро, на цената на по-нестабилно изпълнение;
- направление за развитие - използва се от програмистите докато работват най-новите промени по кода.

SCM предоставят възможности за управление на отделните направления, като по този начин може да се разграничи кои версии от историческото развитие се използват от програмисти, тестери и обикновени потребители.

2 История и развитие на SCM

2.1 Ръчно управление

2.2 Локално управление

2.3 Централизирано управление

2.4 Дистрибутирано управление

3 Сравнение на архитектурно ниво

3.1 Архитектура на централизираните системи

3.1.1 Архитектура на SVN

3.2 Архитектура на дистрибутираните системи

3.2.1 Архитектура на Mercurial

3.2.2 Архитектура на Git

4 Сравнение на функционално ниво

4.1 Предимства на централизираните системи

4.2 Недостатъци на централизираните системи

4.3 Предимства на дистрибутираните системи

4.4 Недостатъци на дистрибутираните системи

5 Заключение

Литература

- [1] <http://example.com/>
- [2] <http://ftp.newartisans.com/pub/git.from.bottom.up.pdf>