

Використання ВПЗ в процесі розробки ПЗ

Подібка І. О.

Львівський національний університет імені Івана Франка, pavione07@gmail.com

The paper describes which free instruments can be used during development process and for what this instruments is used. There is a large number of such software to measure code quality, collect project metrics, build code coverage statistics, manage source code etc. And all this instruments distributed under free-like licences.

Світ вільного програмного забезпечення є дуже великий, а у сфері розробки програмних продуктів він куди більший, ніж ми це собі можемо уявити. Процес розробки програмного забезпечення на усіх рівнях потребує використання різного, подекуди досить специфічного програмного забезпечення. В цій доповіді більшу увагу ми зосередимо на використанні вільного програмного забезпечення в процесі кодування проекту. Інструменти, які будуть згадуватись нижче, є такими, які виправдали себе в роботі і використовуються в повсякденній роботі.

Система керування версіями (source code managment). Найпопулярнішими інструментами є звісно git, svn і mercurial. Найчастіше в розробці ПЗ зустрічається git. Він поширюється за ліцензією GNU GPL v2. І є досить популярні сервіси, такі як GitHub і Bitbucket. Обидва сервіси дають змогу створювати власні репозиторії коду безкоштовно в необмеженій кількості. Але кожен з цих сервісів має свою специфіку щодо надання доступу і кількості користувачів, які мають доступ до одного репозиторію. Більш цікавим варіантом є GitLab. По суті, GitLab є клоном GitHub, але відрізняється тим, що він поширюється безкоштовно і є можливість запустити його на власних серверах і мати необмежені можливості з точки зору можливостей масштабування.

Continuos Integration (CI). Процес автоматизування збірки і компілювання проекту є важливим етапом під час розробки ПЗ. Основною ідеєю цього підходу є те, що побудова проекту відбувається на спеціально налаштованому сервері, який крім того виконує роль тестування ПЗ - запускає unit-тести, інтеграційні тести, збирає статистику проекту тощо. Найпопулярнішими інструментами є TeamCity, Jenkins і Hudson. Кожен з цих інструментів може виконувати побудову проекту під будь-які платформи і не є залежним від мови програмування, оскільки вони лише запускають на виконання компілятори, які встановлені на поточній операційній системі. Найбільш зручним з точки зору використання є TeamCity. Його перевагою є можливість встановлення окремих агентів (build agent) на різні робочі станції, які виконують платформно-залежні побудови проектів. Крім того гнучкість налаштування, можливість встановлення власних модулів для розширення функціональності роблять TeamCity зручним інструментом з впровадження

CI на проекті.

Інструменти для статистики покриття коду (code coverage).

Важливою частиною розробки коду є його аналіз на покриття тестами, тобто оцінити який відсоток коду був виконаний після виконання відповідних автоматичних тестів. Це є дуже важливим аспектом, оскільки дає можливість побачити, які частини коду були виконані, а для яких потрібно ще створити відповідні тести. Таких інструментів є дуже багато, більшість з них залежні від мови програмування, тому все залежить від специфіки проекту. Наведемо короткий список open-source проектів:

№	Java	.Net	C/C++	Perl	Python
1	EMMA	OpenCover	tcov	Devel::Cover	coverage.py
2	Hansel	NCover	gcov	Pod::Coverag	pylid
3	jcoverage	Clover.NET	Squish Coco	Test::Pod::Coverage	trace.py
4	Cobertura		OC		
5	Clover				
6	GroboUtils				
7	Quilt				
8	SonarQube				

Найбільш цікавим є проект SonarQube. Це ціла система, яка може виконувати різнобічний аналіз коду, генерувати різноманітні звіти, знаходити потенційні проблеми в проекті тощо. SonarQube підтримує більш ніж 25 мов програмування, а також інтегрується з різними середовищами розробки.

Щодо середовищ розробки, то тут робити аналіз вільнопоширюваних програмних продуктів досить важко, оскільки все залежить від проекту і вибраної платформи/мови програмування.

Резюмуючи вищесказане, можна зробити висновок, що вільних чи open-source інструментів, які допомагають в розробці ПЗ є дуже багато, а їх якість є досить високою. Це пояснюється тим, що в розвитку процесів розробки ПЗ були створені інструменти, які пришвидшують цей процес, тобто такі інструменти були створені розробниками для розробників.

Література

1. <http://c2.com/cgi/wiki?CodeCoverageTools>
2. http://en.wikipedia.org/wiki/List_of_tools_for_static_code_analysis
3. http://en.wikipedia.org/wiki/Continuous_integration
4. <http://en.wikipedia.org/wiki/SonarQube>
5. http://en.wikipedia.org/wiki/Revision_control