

—
—
—
—
—

Д. А. Курлюк, Н. А. Жияк
(БГТУ, Минск)

РАСПРЕДЕЛЕННАЯ СИСТЕМА КОНТРОЛЯ ВЕРСИЙ – GIT

Распределенная (без единого сервера) система контроля версий Git дает массу возможностей программистам для изменения, дополнения и отслеживания изменений в программных продуктах.

У каждого разработчика, использующего Git, есть свой локальный репозиторий, позволяющий локально управлять версиями. Затем, сохраненными в локальный репозиторий данными, можно обмениваться с другими пользователями. Часто при работе с Git создают центральный репозиторий, с которым остальные разработчики синхронизируются. Пример организации системы с центральным репозиторием – это проект разработки ядра Linux’a .

Достоинства распределенной системы следующие:

- надежная система сравнения ревизий и проверки корректности данных, основанные на алгоритме хеширования SHA1;
- гибкая система ветвления проектов и слияния веток между собой;
- наличие локального репозитория, содержащего полную информацию обо всех изменениях, позволяет вести полноценный локальный контроль версий и заливать в главный репозиторий только полностью прошедшие проверку изменения;
- множество графических оболочек, позволяющих быстро и качественно вести работы с Git’ом;
- возможность делать контрольные точки, в которых данные сохраняются без дельта компрессии, а полностью. Это позволяет уменьшить скорость восстановления данных, так как за основу берется ближайшая контрольная точка, и восстановление идет от нее. Если бы контрольные точки отсутствовали, то восстановление больших проектов могло бы занимать часы;
- широкая распространенность, легкая доступность и качественная документация;
- универсальный сетевой доступ с использованием протоколов http, ftp, ssh и др.

Недостатки распределенной системы следующие:

- Unix – ориентированность. На данный момент отсутствует зрелая реализация Git, совместимая с другими операционными системами;
- возможные (но чрезвычайно низкие) совпадения хеш-кода отличных по содержанию ревизий;
- не отслеживается изменение отдельных файлов, а только всего проекта целиком, что может быть неудобно при работе с большими проектами, содержащими множество несвязных файлов;
- при начальном создании репозитория и синхронизации его с другими разработчиками, потребуется достаточно длительное время для скачивания данных, особенно, если проект большой, так как требуется скопировать на локальный компьютер весь репозиторий.

Выводы:

Git – гибкая, удобная и мощная система контроля версий.