

Управление версиями

Выполнили:

Артемьев Ю.

Кондратьев В.

Нестерова Е.

Введение в управление версиями

- ▶ отслеживания истории своих изменений
- ▶ возможность возвратить код к предыдущему более стабильному состоянию
- ▶ все системы управления версиями основываются на механизме разниц (diffs)

Общие сведения

Многие системы управления версиями предоставляют ряд других возможностей:

- ▶ Позволяют создавать разные варианты одного документа, т.н. ветки, с общей историей изменений до точки ветвления и с разными — после неё.
- ▶ Дают возможность узнать, кто и когда добавил или изменил конкретную строку в файле.
- ▶ Ведут журнал изменений, в который пользователи могут записывать информацию о том, что и почему они изменили в данной версии.
- ▶ Контролируют права доступа пользователей, разрешая или запрещая чтение или изменение информации, в зависимости от того, кто запрашивает это действие.

Используемые термины

- ▶ branch
- ▶ check-in, commit, submit
- ▶ check-out
- ▶ conflict
- ▶ merge, integration
- ▶ repository
- ▶ Revision, tag, label
- ▶ update, sync
- ▶ working copy

Распределённые системы

- ▶ отсутствие центрального репозитория, к которому обращаются клиентские программы
- ▶ Отдельные распределённые системы, как Vazaar-NG или Darcs упрощают механизм копирования репозитория по максимуму
- ▶ Возражение, которое нередко выдвигают противники такой модели - лёгкость установки своего репозитория стимулирует форки проектов.
- ▶ Вы вносите изменения и "отправляете" их в репозиторий находясь без подключения к сети

Примеры распределенных систем

- ▶ **Git** — распределённая система контроля версий файлов и совместной работы.
- ▶ **Mercurial** — распределённая система управления версиями, разработанная для эффективной работы с очень большими репозиториями кода.
- ▶ **Microsoft Visual SourceSafe (Visual SourceSafe, VSS)** — файл-серверная система управления версиями, предназначенная для небольших команд разработчиков.
- ▶ **Perforce** — коммерческая система управления версиями.
- ▶ **Rational ClearCase** — система управления версиями разрабатываемая подразделением Rational Software компании IBM.

Централизованные системы

- ▶ Concurrent Versioning System.). Эта система издавна пользуется популярностью, ввиду своей открытости (свободная лицензия GPL, хотя и не с начала разработки, но по сию пору), стабильности и многоплатформенности. Под неё написано огромное количество сервисных программ, графических клиентов и т.п.
- ▶ **Subversion** разработана специально для замены CVS, самой распространённой открытой системы управления версиями. Она обладает всеми основными функциями CVS (хотя некоторые из них выполняет другими способами) и свободна от ряда её недостатков.

CVS vs Subversion

- ▶ Subversion разрабатывалась как продвинутая замена CVS, изначально ставилась задача: сохранить всю привычную функциональность при добавлении новых возможностей.
- ▶ Subversion поддерживает режим работы атомарного принятия изменений
- ▶ Subversion также «обучается» клонировать данные из текущего в произвольный удаленный репозиторий
- ▶ Права доступа к репозиторию
- ▶ Комментарии к каждому измененному объекту
- ▶ Простота развертывания системы
- ▶ Простота интеграции в существующую инфраструктуру сети