

Задание для студентов гр. ИС - 31 на 30.10.2020

Дисциплина «МДК 05.02 Разработка кода информационных систем»

Преподаватель Токарская М.С.

Почта для обратной связи: maya_tok@mail.ru

Тел. 89147174421 – WhatsApp

Задание:

1. Составить конспект по данной теме
2. Письменно выполнить задание в конце темы

Выполненное задание выслать на почту 30.10.2020

Лекция_ Система контроля версий.

Когда активно работаешь с файлами, будь то исходные тексты программ, файлы конфигурации, или статья, то в большинстве случаев нужно использовать какую-либо систему контроля версий.

Основная работа с файлами — это их изменение, то наличие возможности вернуться к предыдущей версии дает огромную уверенность, что в свою очередь положительно влияет на производительность. Если работа идет в команде, то система контроля версий просто необходима, в противном случае, в какой-то момент времени работа команды может просто застопориться.

Система контроля версий (Version Control System, VCS) представляет собой программное обеспечение, которое позволяет отслеживать изменения в документах, при необходимости производить их откат, определять, кто и когда внес исправления и т.п.

Что такое система контроля версий?

Наверное, всем знакома ситуация, когда при работе над проектом, возникает необходимость внести изменения, но при этом нужно сохранить работоспособный вариант, в таком случае, как правило, создается новая папка, название которой скорее всего будет “Новая папка” с дополнением в виде даты или небольшой пометки, в нее копируется рабочая версия проекта и уже с ним производится работа. Со временем количество таких папок может значительно возрасти, что создает трудности в вопросе отката на предыдущие версии, отслеживании изменений и т.п. Эта ситуация значительно ухудшается, когда над проектом работает несколько человек.

Для решения таких проблем как раз и используется система контроля версий, она позволяет комфортно работать над проектом как индивидуально, так в коллективе. VCS отслеживает изменения в файлах, предоставляет возможности для создания новых и слияние существующих ветвей проекта, производит контроль доступа пользователей к проекту, позволяет откатывать исправления и определять кто, когда и какие изменения вносил в проект.

Основным понятием VCS является **репозиторий (repository)** – специальное хранилище файлов и папок проекта, изменения в которых отслеживаются. В распоряжении разработчика имеется так называемая “**рабочая копия**” (*working copy*) проекта, с которой он непосредственно работает. Рабочую копию необходимо периодически синхронизировать с репозиторием, эта операция предполагает отправку в него изменений, которые пользователь внес в свою рабочую копию (такая операция называется *commit*) и актуализацию рабочей копии, в процессе которой к пользователю загружается последняя версия из репозитория (этот процесс носит название *update*).

Централизованные и распределенные системы контроля версий

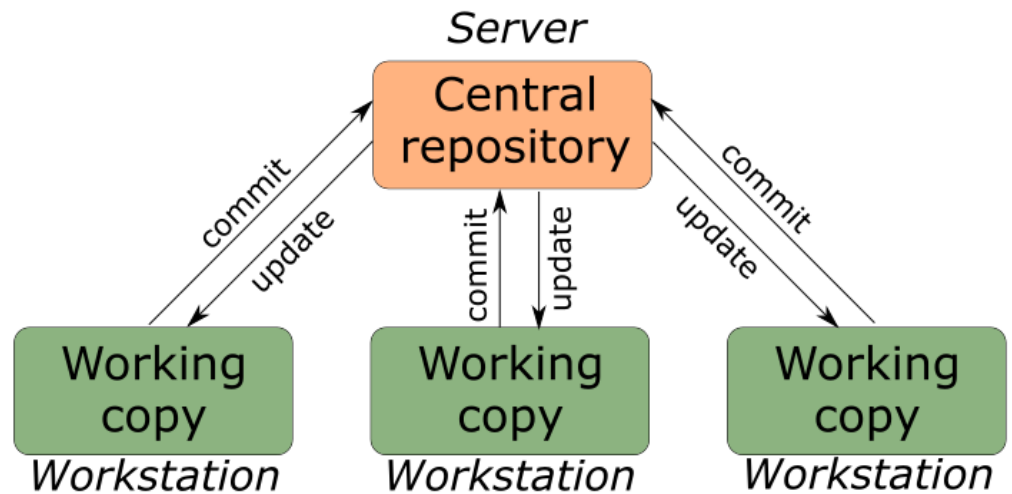
Системы контроля версий можно разделить на две группы: распределенные и централизованные.

Централизованные системы контроля версий

Централизованные системы контроля версий представляют собой приложения типа клиент-сервер, когда репозиторий проекта существует в единственном экземпляре и хранится на сервере. Доступ к нему осуществляется через специальное клиентское приложение.

Примеры:

1. CVS (*Concurrent Versions System*)
2. Subversion (*SVN*)



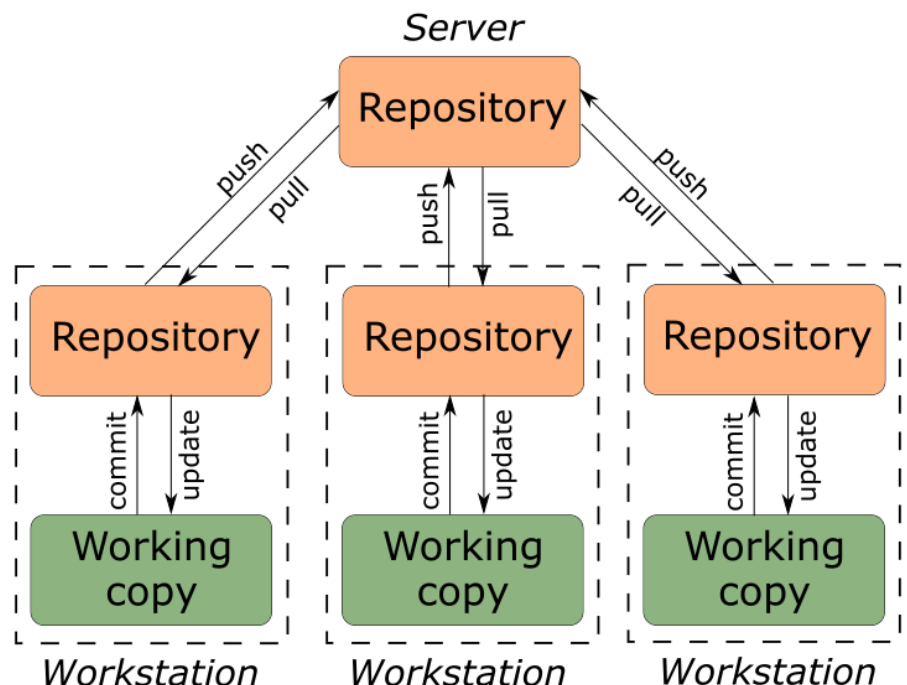
Распределенные системы контроля версий

Распределенные системы контроля версий (*Distributed Version Control System, DVCS*) позволяют хранить репозиторий (его копию) у каждого разработчика, работающего с данной системой. При этом можно выделить центральный репозиторий (условно), в который будут отправляться изменения из локальных и, с ним же эти локальные репозитории будут синхронизироваться.

При работе с такой системой, пользователи периодически синхронизируют свои локальные репозитории с центральным и работают непосредственно со своей локальной копией. После внесения достаточного количества изменений в локальную копию они (изменения) отправляются на сервер. При этом сервер, чаще всего, выбирается условно, т.к. в большинстве DVCS нет такого понятия как “выделенный сервер с центральным репозиторием”.

Большое преимущество такого подхода заключается в автономии разработчика при работе над проектом, гибкости общей системы и повышение надежности, благодаря тому, что каждый разработчик имеет локальную копию центрального репозитория.

Две наиболее известные DVCS – это *Git* и *Mercurial*.



Задания по теме:

Опишите работу VCS (основные элементы, функции, принципы работы):

- a) CVS
- b) Subversion
- c) Git
- d) Mercurial