# РАБОТА №1 ЛАБОРАТОРНАЯ. СОЗДАНИЕ И ИЗУЧЕНИЕ ВОЗМОЖНОСТЕЙ РЕПОЗИТОРИЯ ПРОЕКТА

## 1. ЦЕЛЬ РАБОТЫ

Целью выполнения лабораторных работ является:

 получить опыт практической работы с системой контроля версий на примере Tortoise SVN;  получить навыки создания репозитория.

## 2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Под системой контроля версий понимается механизм сохранения промежуточных состояний кода разрабатываемого программного обеспечения. То есть с помощью этой системы программист может управлять своими файлами во времени: смотреть историю изменений файлов и каталогов, возвращаться к более ранним версиям кода, объединять несколько версий файла. Основной областью применения контроля версий является коллективная разработка чего-либо (чаще всего это программы, но область применения программированием не ограничивается). Однако и для разработчикаодиночки контроль версий может быть полезен.

Для решения этих проблем удобно использовать системы контроля версий, например Subversion (SVN).

Subversion (SVN) – одна из наиболее известных систем версионного управления исходным кодом. Применяется во многих Open Source проектах. Подобные системы позволяют узнать, кто, когда и как именно вносил изменения в файлы проекта. Все это значительно облегчает групповую разработку ПО.

Обычно система контроля версий состоит из двух частей:

Сервер, или репозиторий – где хранятся все исходные коды программы, а также история их изменения.

Клиент. Каждый клиент имеет свою локальную копию

(working copy) исходных кодов, с которой работает разработчик.

В связи с тем, что разработчики работают только с локальными копиями, могут возникать трудности, когда два и более человек изменяют один и тот же файл.

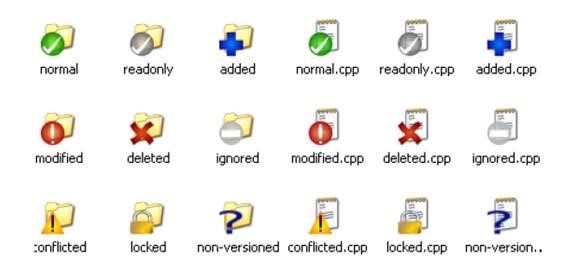
В системе контроля версий есть две модели, которые позволяют избегать этой проблемы:

Блокировка – изменение – разблокировка. Согласно этой модели, когда кто-либо начинает работу с файлом, этот файл блокируется, и все остальные пользователи теряют возможность его редактирования. Очевидным недостатком такой модели является то, что файлы могут оказаться надолго заблокированными, что приводит к простоям в разработке проекта.

Копирование – изменение – слияние. В данной модели каждый разработчик свободно редактирует свою локальную копию файлов, после чего выполняется слияние изменений. Недостаток этой модели в том, что может возникать необходимость разрешения конфликтов между изменениями файла.

Получение информации о статусе

При работе с рабочей копией, вам часто надо понять, какие файлы вы заменили,/добавили,/удалили или переименовали, или же какие из файлов были изменены и зафиксированы другими.



### Рис.1.1 Пометки на значках в системе SVN

Теперь, после извлечения рабочей копии из хранилища Subversion, вы можете заметить, что значки в Проводнике Windows немного изменились, и это одна из причин такой популярности TortoiseSVN. TortoiseSVN добавляет так называемую пометку на значке для каждого файла, которая накладывается на исходный значок файла. Пометки различаются и зависят от статуса файла в Subversion.

 – В свеже-извлечѐнной рабочей копии все пометки выглядят как зеленая галочка. Это означает, что статус Subversion – нормальный.

 – Файлы были изменены с момента последнего обновления \ рабочей копии и нуждаются в фиксации.

 – В процессе обновления возник конфликт (пометка- желтый восклицательный знак).

 – Для файла установлено свойство svn:needs-lock, Subversion (файл как доступный только для чтения). Пометка на файлах означает, что вы должны заблокировать файл перед тем, как начнѐте его редактировать.

 – Вы владеете блокировкой на файл и его статус в Subversion нормальный. Вы должны разблокировать файл, если вы его не используете, чтобы и другие могли зафиксировать свои изменения в этом файле.

 – Пометка папки – некоторые файлы или папки внутри текущей папки запланированы

 – Символ плюс – файл или папка запланированы для добавления под управление версиями.

 – Символ минус – файл или папка игнорируется систе-

мой управления версиями. Это необязательная пометка.

 – файлы, не находятся под управлением версиями, но в то

же время не являются игнорируемыми (необязательная пометка).

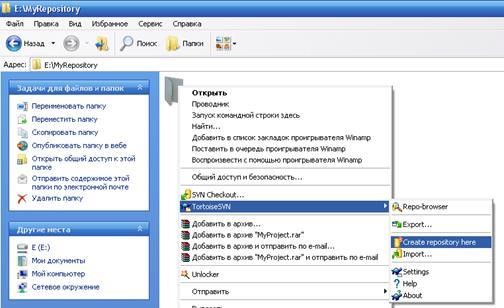
## 3. ПОРЯДОК ВЫПОЛНЕНИЯ И ЗАДАНИЯ

### Задание №1

#### 1. Создание хранилища (репозитория) SVN

1.1. На одном из локальных дисков создайте папку, в которой в дальнейшем будет находится репозиторий SVN для проекта. Лучше, чтобы эта папка находилась на достаточно надежном носителе. Пусть это будет папка E:\MyRepository\MyProject.

1.2. Нажмите правой кнопкой мыши на созданной папке и выберите команду TortoiseSVN\Create repository here.



### Рис.1.2 Создание репозитория

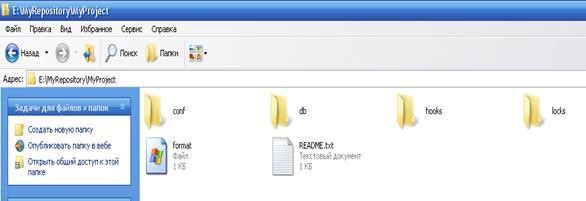
1.3. Появится окно с сообщением, что репозиторий успешно создан. После чего нажмите кнопку ОК.

1.4.



### Рис.1.3 Окно сообщения об успешном создании хранилище

1.5. Далее появятся файлы как показано ниже на рисунке/



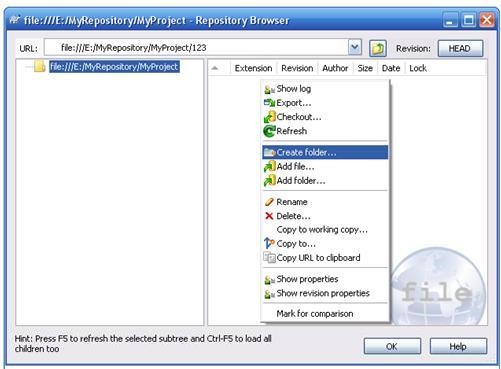
### Рис.1.4 Структура репозитория

*Примечание: Хранилище будет создано внутри новой папки. Не редактируйте эти файлы самостоятельно!!! В случае возникновения ошибок убедитесь, что папка пуста и доступна для записи.*

1. Импорт данных в хранилище

Предполагая, что хранилище уже у вас есть, и вы желаете добавить в него новую папку со всей еѐ структурой, просто выполните следующие шаги:

2.1. При помощи обозревателя хранилища (TortoiseSVN → Repo-browser) создайте новую папку проекта под именем **123** непосредственно в хранилище.



### Рис.1.5 Обозреватель хранилища

2.2. Откройте общий доступ хранилище вкладе Доступ пометив флажком в Открыть общий доступ к этой папке и Разрешить изменение файлов в сети.

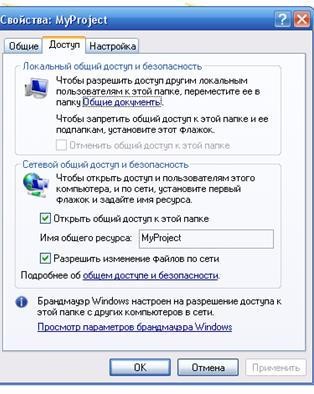
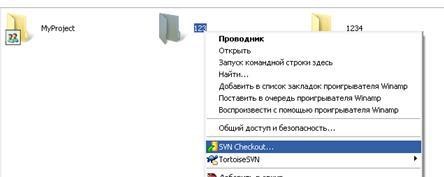


Рис.1.6 Открытие общего доступа к репозиторий.

1. Извлечение рабочей копии

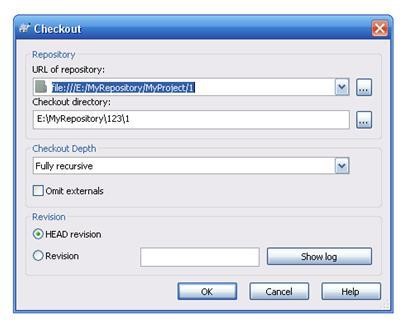
3.1. Для получения рабочей копии вам надо произвести извлечение из хранилища. Выберите в Проводнике Windows папку, в которой хотите поместить вашу рабочую копию. Сделайте правый щелчок для вызова контекстного меню и выберите команду

TortoiseSVN -> Извлечь ...,(TortoiseSVN → Checkout..., )



### Рис.1.7 Меню для извлечения рабочей копии

3.2. Появившемся диалоговом окне Checkout в поле URL of Repository вставьте путь к репозиторий и нажимаем ОК, далее еще раз нажимаете ОК. Для того что вставить выделите репозиторий – папку и с помощью правой кнопки мыши откройте обозревателя хранилища (TortoiseSVN → Repo-browser) в поле URL скопируйте путь. В поле Checkout directory автоматически появляется путь к рабочей копии, т. е. к папке.



### Рис.1.8 Путь к репозиторий

3.3. Выделите рабочую копию, т. е. папку под именем **123** и нажмите F5, либо правой кнопкой мыши нажмите обновить.



Рис.1.9 Свеже-извлечѐнная рабочая копия

3.4. Повторите действия 3.1–3.3 для рабочей копии **1234**.

### Задание №2

1. Аналогично проделать те же действие, но проделав эти же действие на двух разных компьютерах.
2. В одном компьютере например, комп1 создаете репозиторию и одну рабочую копию, т. е. папку под именем **123.**
3. Во втором компьютере например, комп2 создаете только рабочую копию, т. е. папку под именем **1234.**
4. Во втором компьютере соедините рабочую копию с репозиторием указывая путь file:///ip-компьютера/название\_репозитория.
5. В комп1 создайте любой файл и отправьте его на комп2. Таким образом проделав 5 ревизий.
6. Чтобы узнать IP- компьютера нажмите пуск →выполнить→ cmd, либо

C:\Windows\System32\cmd.exe /k %windir%\system32\ipconfig.exe

1. Откроется окно командой строки. Введите ipconfig, где вы увидите примерное IP- компьютера: 192.168.8.xxx

## 4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Что такое Хранилище (Repository)?
2. Что такое Рабочая копия (Working Copy)?
3. Что такое версия?
4. Понятия управления контроля версиями, применение контроля версий.
5. Чем помогают системы контроля версий типа SVN?