

Основы программной инженерии

Контроль версий в программных проектах

2017

Групповая работа с проектом

► Основные задачи:

- Повышение надежности хранения артефактов
- Общий доступ к файлам
- Сохранение истории модификации файла
- Возможность возврата к предыдущим версиям
- Пометка отдельных версий файла
- Поддержание и развитие нескольких параллельных историй артефакта
- Сравнение версий
- Объединение разрозненного кода

Общий доступ к файлам

- ▶ Одновременное редактирование одного файла разными пользователями
 - ▶ Потеря изменений, сделанных пользователем (затерты записью изменений другого пользователя)
- * В параллельных системах для разрешения используются семафоры, мьютексы, критические секции и т.п.

Сохранение истории модификации файла

- ▶ Необходимо иметь историю изменения файла:
 - Версия
 - Автор изменения
 - Время изменения
 - Суть изменения
 - Причина изменения
 - И т.п.

Пометка отдельных версий файла

- ▶ Отдельные версии файлов необходимо специально пометить
- ▶ Причина пометки:
 - Качественная версия
 - Версия, обладающая определенными свойствами
 - Версия, являющаяся частью релиза проекта определенной версии
 - ...

Поддержание и развитие нескольких историй файла

- ▶ Причины разветвления версий файла:
 - Развитие нескольких версий проекта
 - Поставленных заказчику
 - Разрабатываемых
 - Наличие нескольких конфигураций проекта
 - Для разной аппаратуры
 - Для разных операционных систем
 - Разработка новой (экспериментальной) функциональности
 - ...

Системы контроля версий

▶ Другие названия:

- Системы управления версиями (VCS - Version Control System)
- Системы контроля ревизий (RCS - Revision Control System)
- Системы управления исходным кодом (SCM – Source Code Management)

Системы контроля версий

- ▶ СКВ предназначены для автоматизации групповой работы и управления версионированием проектов
- ▶ СКВ обеспечивают
 - Репозиторий (или репозитории) хранения проектов
 - Стандартные операции обеспечения групповой работы
 - Клиенты для выполнения операций

Типы СКВ

- ▶ **Централизованные СКВ**
 - Единое централизованное хранилище
 - Клиент-серверный доступ
 - Примеры: CVS, Subversion, MS Visual Source Safe и т.п.
- ▶ **Распределённые СКВ**
 - Репозиторий хранится на каждом компьютере
 - Сетевая синхронизация репозиториях посредством слияний (заплаток, патчей, change sets и т.п.)
 - Используется в интернет-проектах, когда разработчики существенно удалены друг от друга
 - Примеры: mercurial, git, Bazaar и т.п.

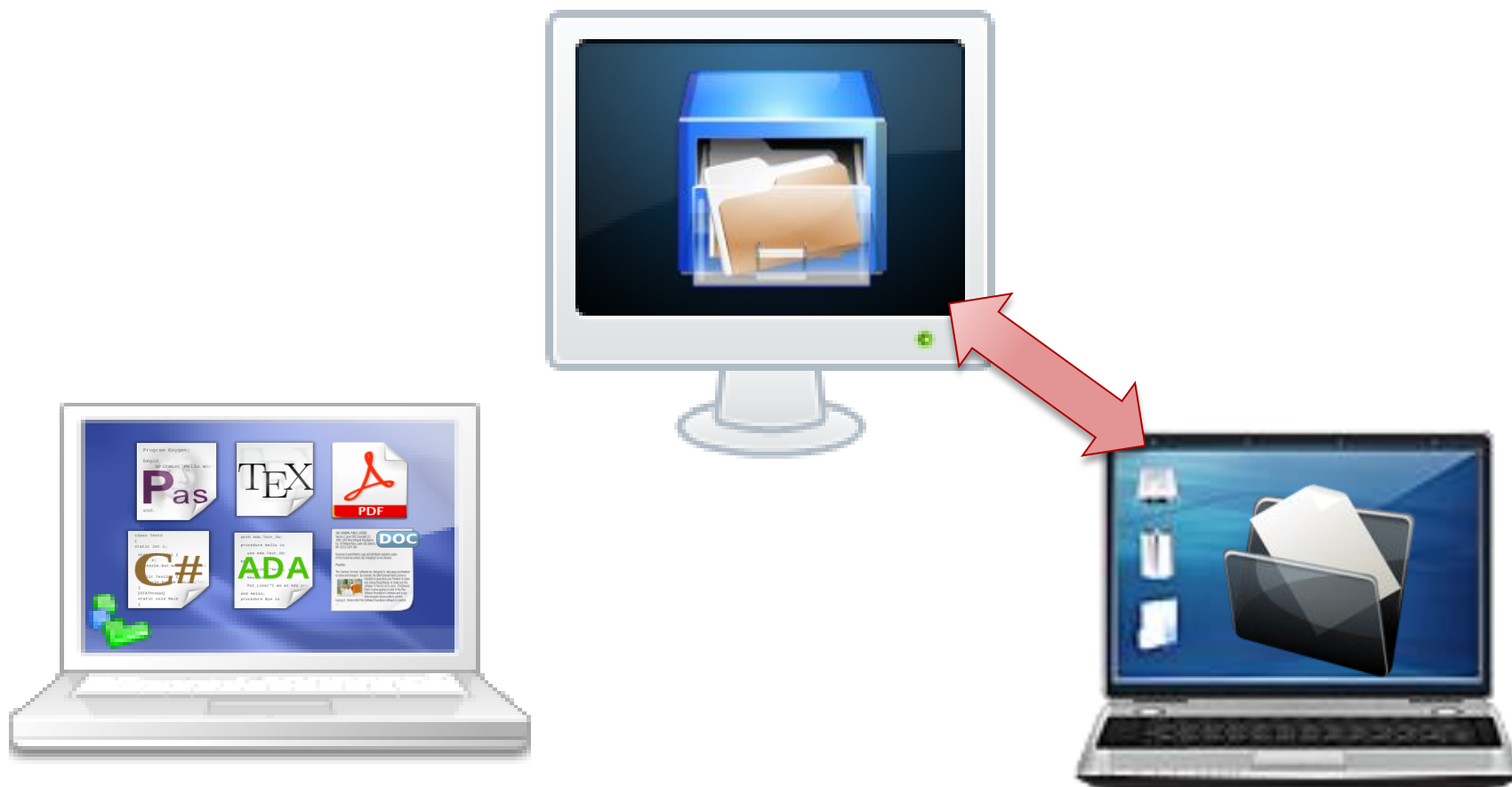
Ревизия файла

- ▶ Ревизия файла – уникальный идентификатор версии файла в системе контроля версии
 - CVS: 1.2
 - SVN: 238
 - Git, Mercurial: хэш SHA-1
- ▶ При изменении файла номер ревизии изсеняется определенному правилу
 - иногда номер ревизия является атрибутом всего репозитория
- ▶ Атрибуты ревизии
 - Идентификатор
 - Автор изменения
 - Дата
 - Текстовое описание изменения
 - Внешние атрибуты
 - Тэги
 - Идентификаторы ветвей

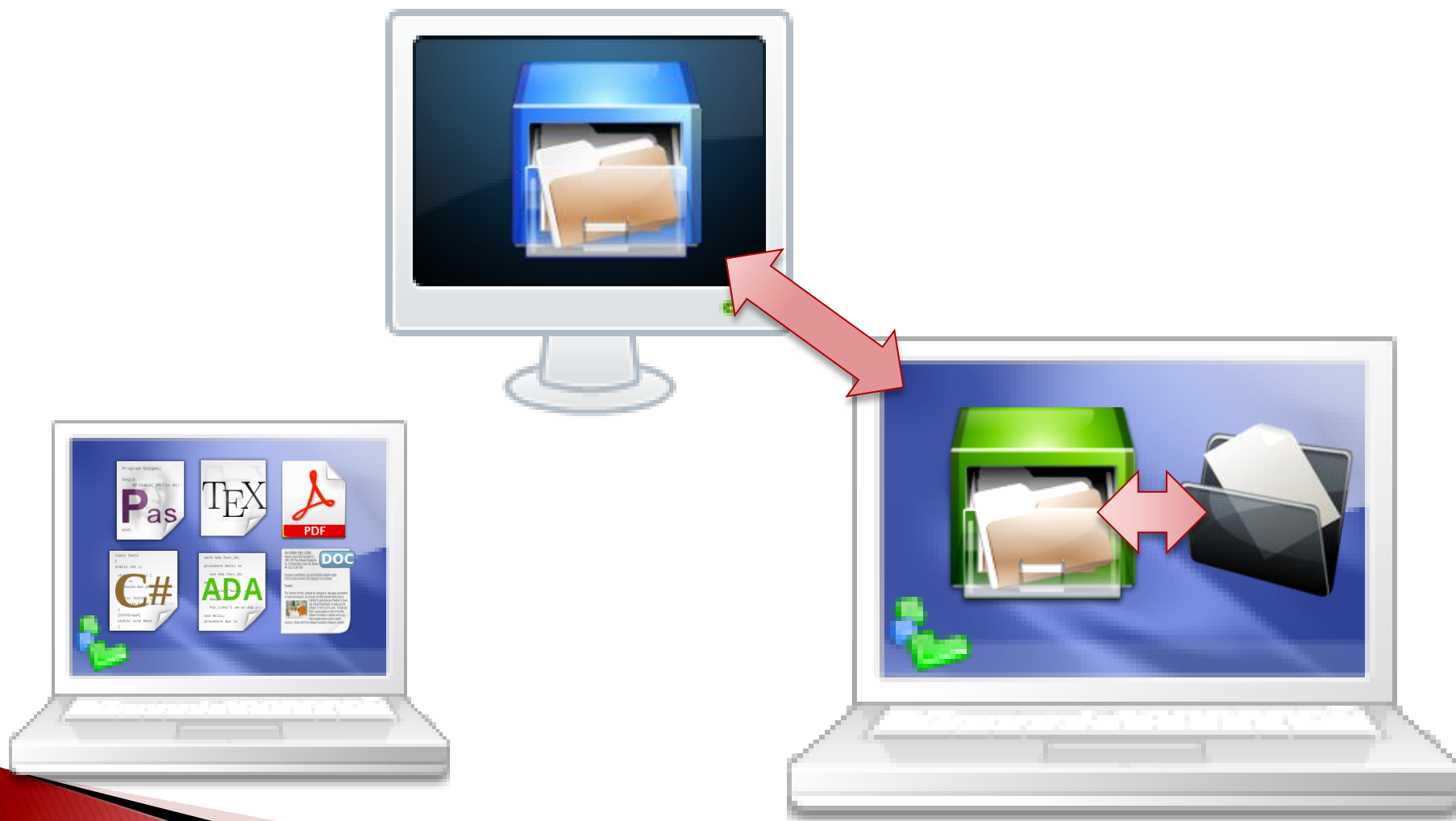
Хранимые копии проекта

- ▶ Для централизованных СКВ:
 - Локальная копия проекта
 - Локальная копия проекта, находящегося под контролем СКВ
 - Серверная копия, находящаяся в репозитории
- ▶ Для распределенных СКВ:
 - Локальная копия проекта
 - Локальная копия проекта, находящегося под контролем СКВ
 - Копия, находящаяся в локальном репозитории
 - Копия, находящаяся в удаленном репозитории

Централизованные СКВ



Распределенные СКВ



Общие принципы хранения файлов в системах версионирования

- ▶ Поддержка текстового и бинарного формата хранения
- ▶ Для текстового формата:
 - Хранение инкрементных изменений
 - Возможность визуального сравнения ревизий
- ▶ Для бинарного формата:
 - Хранение полных версий артефактов

Пометка версий в СКВ

- ▶ Основные способы
 - Тэги
 - Виртуальные каталоги

Тэги

- ▶ Тэг – текстовая метка, привязанная к какой-либо ревизии файла или репозитория
- ▶ Одна ревизия может содержать несколько тэгов
- ▶ Выборку ревизии файла/файлов можно производить по тэгам

Виртуальные каталоги

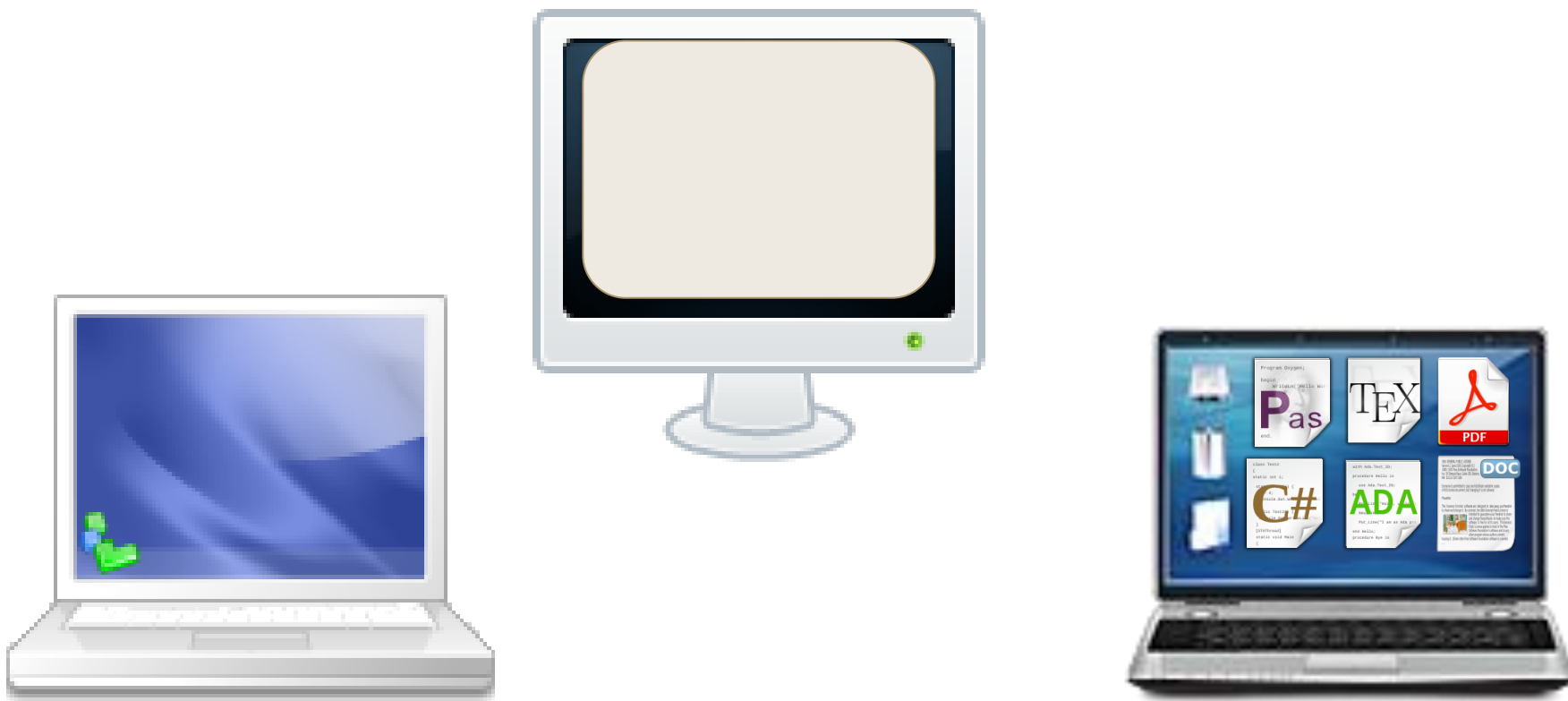
- ▶ Вместо тэга создается каталог
- ▶ В него помещаются виртуальные копии необходимых ревизий всех требуемых файлов
- ▶ Работа с таким каталогом происходит стандартными способами

Операции в системе контроля версий

- ▶ Импорт проекта (import)
- ▶ Экспорт проекта (export)
- ▶ Получение проекта (checkout, clone, ...)
- ▶ Обновление файла (update)
- ▶ Фиксация изменений (commit)
- ▶ Сравнение изменений
- ▶ Установка тэгов
- ▶ Переход у другой ревизии (откат)
- ▶ Создание ветвей
- ▶ Переключение на ветвь
- ▶ Слияние (merge)
- ▶ Разрешение конфликтов
- ▶ Блокировка файлов
- ▶ Выгрузка изменений (push)
- ▶ Запрос на изменение (pull request)
- ▶ ...

Импорт проекта (import)

Первоначальное помещение локального проекта в репозиторий



Экспорт проекта (export)

- Извлечение проекта из СКВ в локальный каталог
- Удаление проекта из СКВ



Получение проекта (checkout, clone)

- ▶ Получение локального слепка проекта
- ▶ Получение осуществляется по одному из критериев:
 - Головная версия (HEAD, trunk, default, master, ...)
 - Версия на определенную дату
 - Версия с определенным тэгом
 - Версия из определенной ветви
 - ...

Получение проекта



Фиксация изменения (commit)

- Посылка измененной версии файла в репозиторий
- Операция игнорируется, если ревизия на сервере изменилась



Обновление файлов (update)

- Копирование свежей версии из репозитория
- Слияние локальных изменений и серверных в локальном файле

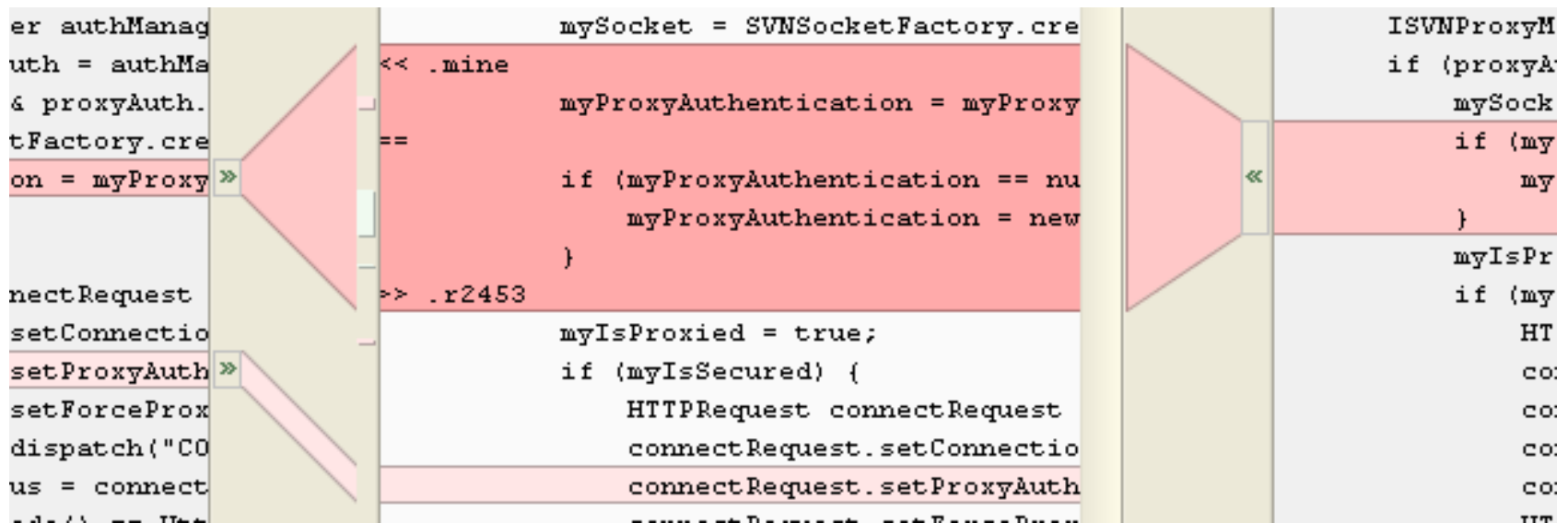


Операции в СКВ. Разрешение конфликтов

- ▶ Конфликт – когда одна и та же строка была по-разному изменена в двух версиях
- ▶ Если при слиянии произошел конфликт – в текст попадают обе версии участков кода с пометками
- ▶ Разрешение проводится **только** в локальной копии
- ▶ В репозитории хранятся только утвержденные версии с разрешенным конфликтом

* Отсутствие конфликта не означает отсутствия ошибок!

Операции в СКВ. Разрешение конфликтов



Операции в системе контроля версий

- ▶ Сравнение изменений
 - Действует только для текстовых файлов
 - Сравнить можно любые две ревизии одного файла из любых ветвей проекта

Операции в системе контроля версий. Сравнение ревизий

CVS Diff - Communication.pas

Исправление1.42

```
1248 ASender.Reply.Text.Add(  
1249   ExchangeClassName(ExchangeList[i].ExchangeClass)  
1250   IntToStr(ExchangeList[i].ExchangeNumber) + ' (' +  
1251   ExchangeList[i].Name + ') ' + ThreadState(ExchangeLi  
1252   //   ' : HndState: ' + HandlerState(ExchangeList[i].Handl  
1253   //   ' : CmdState: ' + HandlerState(ExchangeList[i].Comm  
1254   ' : State: ' + ExchangeState(ExchangeList[i]));  
1255 end;  
1256 for i := 0 to CommandList.Count - 1 do  
1257   ASender.Reply.Text.Add('Command ' + IntToStr(Comman  
1258   ThreadState(CommandList[i].FCommandHandlerInMain  
1259   ' : HndState: ' + HandlerState(CommandHandlerEvent  
1260   ' : CmdState: ' + HandlerState(CommandCommandEve  
1261 // ASender.Thread.Connection.Disconnect;  
1262 end;  
1263 procedure TCommunication.ResetCommand(ASender: TIdC  
1264 begin  
1265   LogMsg('RESET command received');  
1266   ASender.PerformReply := True;  
1267   ASender.Reply.NumericCode := 200;  
1268   ASender.Reply.Text.Add('Success');  
1269   ResetCommunication;  
1270 end;
```

Исправление1.41

```
1208 ASender.Reply.Text.Add(  
1209   ExchangeClassName(ExchangeList[i].ExchangeClass)  
1210   IntToStr(ExchangeList[i].ExchangeNumber) + ' (' +  
1211   ExchangeList[i].Name + ') ' + ThreadState(ExchangeLi  
1212   |   ' : HandlerState: ' + HandlerState(ExchangeList[i].Han  
1213   |   ' : CommandState: ' + HandlerState(ExchangeList[i].Cc  
1214   +++++ -  
1215 end;  
1216 for i := 0 to CommandList.Count - 1 do  
1217   ASender.Reply.Text.Add('Command ' + IntToStr(Comman  
1218   |   ThreadState(CommandList[i].FCommandHandlerInMain  
1219   |   ' : HandlerState: ' + HandlerState(CommandHandlerEx  
1220   |   ' : CommandState: ' + HandlerState(CommandComm  
1221 // ASender.Thread.Connection.Disconnect;  
1222 end;  
1223 procedure TCommunication.ResetCommand(ASender: TIdC  
1224 begin  
1225   LogMsg('RESET command received');  
1226   ASender.PerformReply := True;  
1227   ASender.Reply.NumericCode := 200;  
1228   ASender.Reply.Text.Add('Success');  
1229   ResetCommunication;  
1230 end;
```

☒ Синхронизировать прокрутку измененные блоки 34 измененные строки 73 << Назад Вперед >>

Close

Ветви файлов

- ▶ Ветвь (ветка, branch) – механизм, который служит для ветвления дерева ревизий файла
- ▶ Имя ветви однозначно определяет группу ревизий (ветвь)
- ▶ Имя ветви используется для переключения между ветвями ревизий файла

Системы контроля версий

- ▶ Свободно-распространяемые системы
 - * RCS
 - CVS
 - Subversion (SVN)
 - Mercurial
 - Git
 - Bazaar
 - Darcs
 - ...

Система CVS

- ▶ Одна из первых и изначально самых распространенных систем
- ▶ Состоит из двух частей: сервера и клиента
- ▶ Стандартный клиент – консольный, позволяет выполнить все операции
- ▶ Имеется множество графических клиентов
- ▶ <http://www.cyclic.com/>

Система CVS

LinCVS for Windows - Version 1.3.2

Проект Каталог Файл Параметры Помощь

Файлы CVS Не CVS файлы Игнорируемые файлы

Имя	Исправление	Отметка	Параметры	Состояние	Last updated	Изменения (локально)
AutoServer.pas	1.6			seems up to date	2004/06/16 16:38:40	
AutoServer.xml	1.4			seems up to date	2004/06/21 16:45:50	
AutoServerProject.dpr	1.1			seems up to date	2004/02/16 16:55:54	
AutoServerProject.res	1.1		-kb	seems up to date	2004/02/16 16:55:02	
Client.pas	1.14			seems up to date	2004/11/06 17:07:12	
Client.xml	1.14			seems up to date	2005/02/28 20:49:24	
ClientProject.cfg	1.2			seems up to date	2004/10/15 17:32:25	
ClientProject.dpr	1.3			seems up to date	2004/10/07 18:45:58	
ClientProject.res	1.3		-kb	seems up to date	2004/01/28 22:10:32	
proba.txt	1.4			modified	2004/10/14 13:26:30 2005/02/28 19:52:04	
proba1.txt	1.1.1.1			seems up to date	2005/02/28 19:52:04	
ProjectGroup1.bpg	1.2			seems up to date	2004/02/16 16:56:34	
Server.pas	1.16			seems up to date	2004/11/06 17:06:52	
Server.xml	1.12			!!check timezone: incorrect!!	2004/11/06 17:04:44	
ServerProject.cfg	1.2			seems up to date	2004/10/27 20:33:09	
ServerProject.dpr	1.3			seems up to date	2004/10/26 18:04:36	
ServerProject.res	1.3		-kb	seems up to date	2004/01/28 22:10:32	
test.txt	1.3			seems up to date	2004/10/27 13:11:38	
test1.txt	1.1.1.1			seems up to date	2004/01/28 22:10:32	
ttt.txt	1.3			seems up to date	2004/06/22 20:10:14	

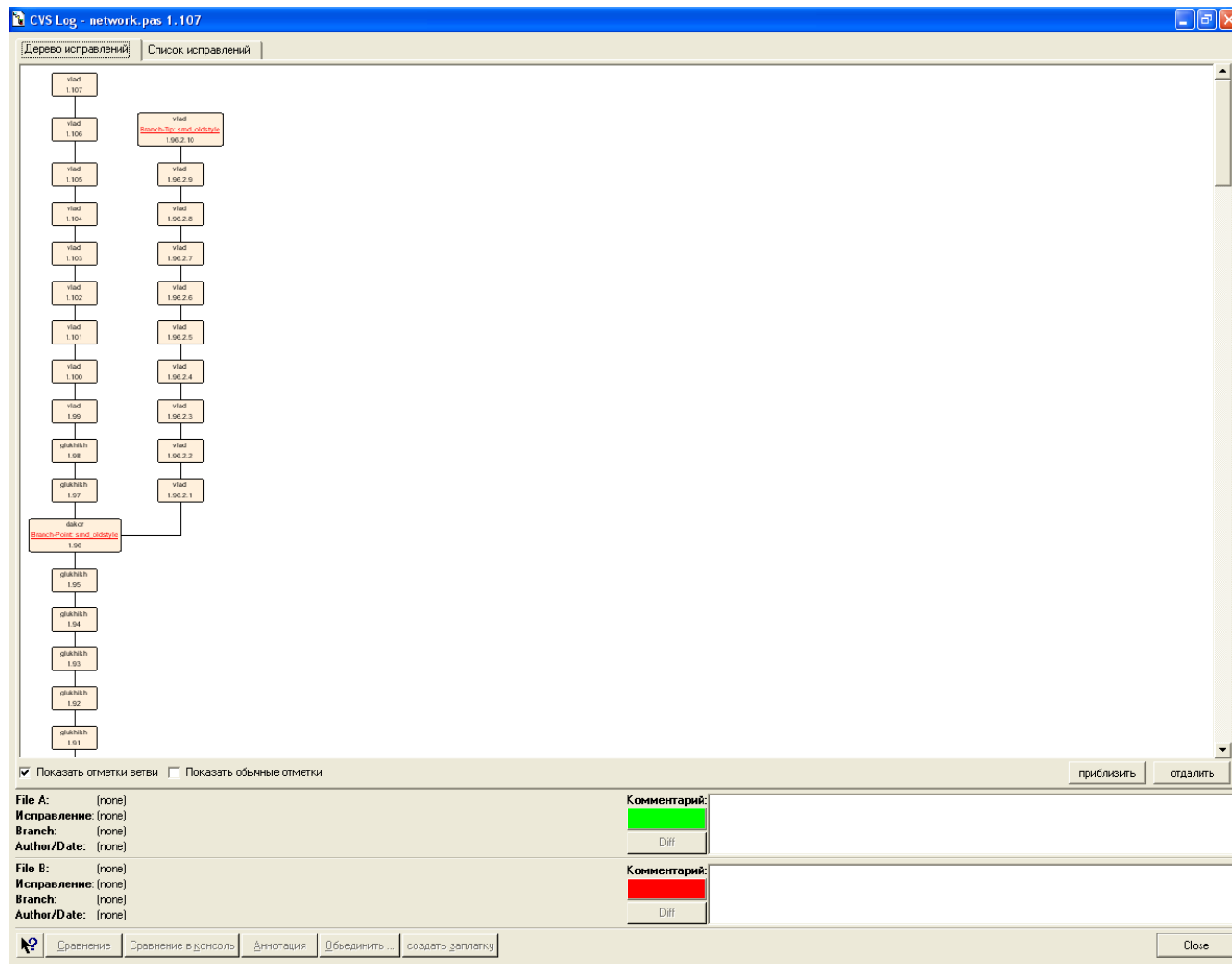
```

end; //if >"
end;
end else begin
  Result := Assigned(OnExecute);
  if Result then begin
    OnExecute(AThread);
  end;
end;
end;
end.

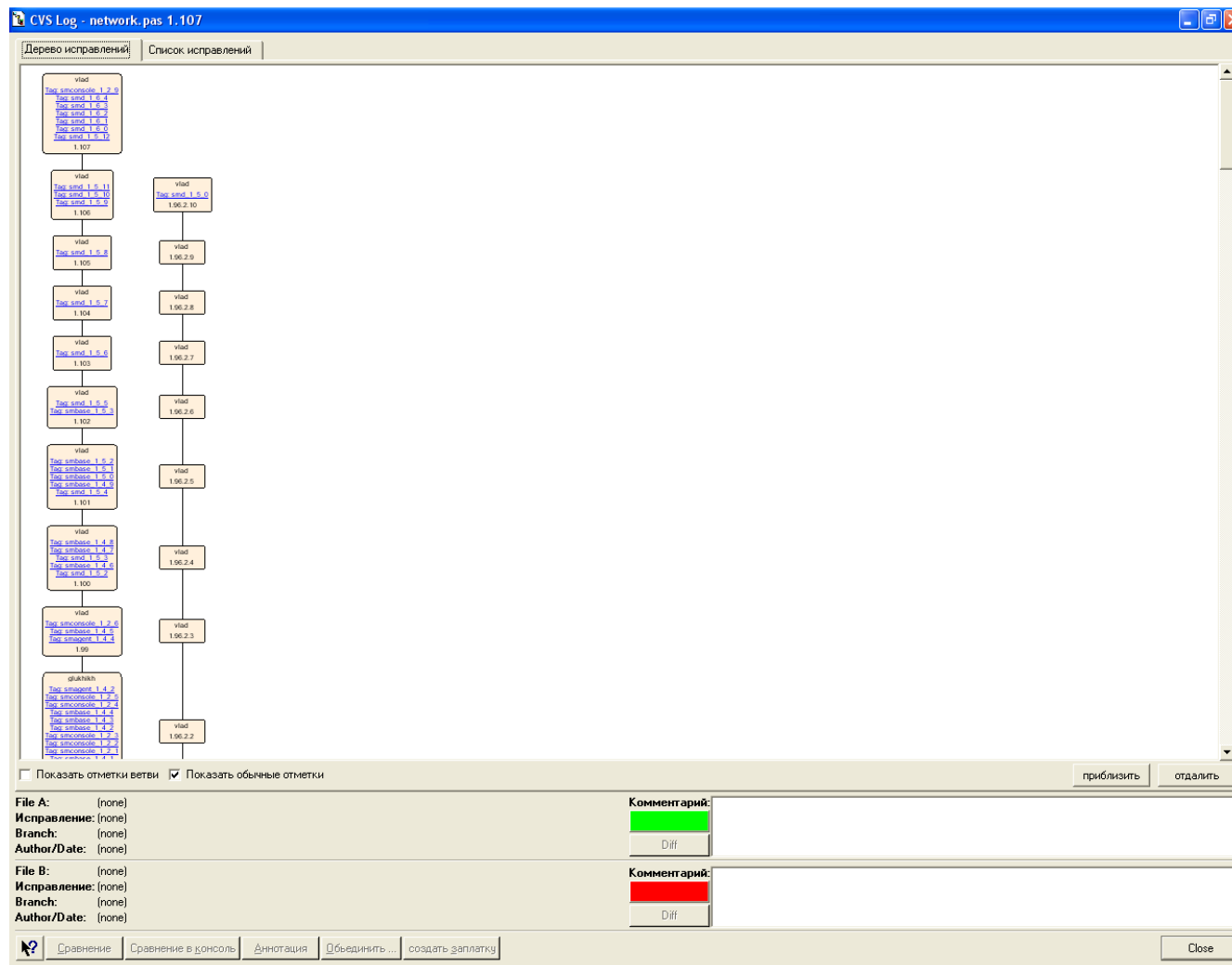
end; //if >"
end;
end else begin
  Result := Assigned(OnExecute);
  if Result then begin
    OnExecute(AThread);
  end;
end;
end;
end.
    
```

E:\job\Digitek\Components\DigitekNetworks\Demo

Система CVS



Система CVS



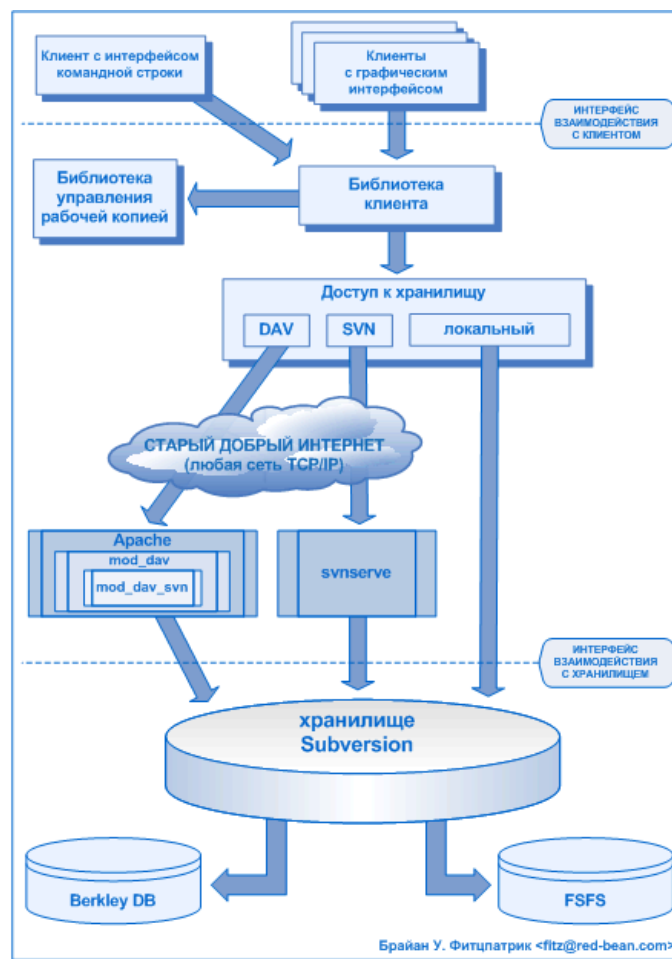
Система Subversion (SVN)

- ▶ Создана с целью заменить CVS
- ▶ Исправляет большинство недостатков CVS
- ▶ <http://subversion.tigris.org>
- ▶ Имеется большое количество графических клиентов

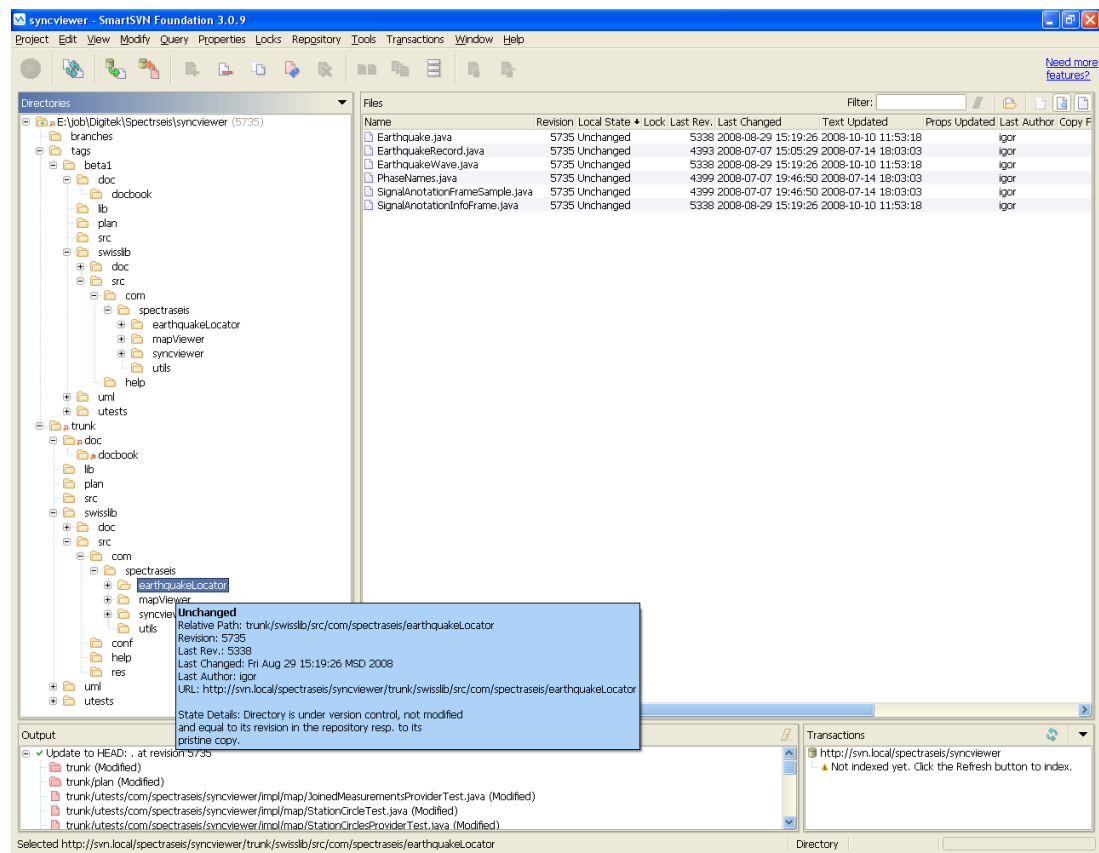
Система Subversion

- ▶ Основные особенности:
 - Замена тэгов и ветвей на виртуальные каталоги
 - Версионирование каталогов
 - Версионирование метаданных
 - Атомарная фиксация изменений
 - Полноценная история версий (удаления, переименования и т.п.)
 - Хранение файлов в сжатом виде
 - Выбор способа доступа к репозиторию
 - Интеграция с web-сервером

Система Subversion



Система SmartSVN



Распределенные СКВ

- ▶ Mercurial
- ▶ Git
- ▶ Bazaar
- ▶ Darcs
- ▶ Monotone
- ▶ GNU arch
- ▶ ...

Коммерческие системы контроля версий

- ▶ Microsoft Visual SourceSafe
- ▶ IBM Rational ClearCase
- ▶ Borland StarTeam
- ▶ Perforce
- ▶ Microsoft Team Foundation Server
- ▶ ...