

ОРГАНИЗАЦИЯ ПРОЦЕССОВ РАЗРАБОТКИ ПО С ПОМОЩЬЮ VISUAL STUDIO 2013 И TFS 2013

<http://www.microsoftvirtualacademy.com/training-courses/software-development-process-with-visual-studio-2013-tfs-2013-rus>

Данный видеокурс будет интересен разработчикам и QA менеджерам. В рамках курса показан обзор возможностей продуктов Visual Studio 2013 и сделан упор на то, чтобы дать слушателям общее понимание возможностей инструментов Visual Studio в процессе разработки ПО и показать, какие выгоды от их использования получит каждая роль процесса разработки ПО и организация в целом. В ходе курса будут рассмотрены возможности инструментов Visual Studio, начиная от планирования проекта разработки и заканчивая его тестированием.

ОРГАНИЗАЦИЯ ПРОЦЕССОВ РАЗРАБОТКИ ПО С ПОМОЩЬЮ VISUAL STUDIO 2013 И TFS 2013.....	1
ВВЕДЕНИЕ.....	3
01 ПРОЦЕССЫ ALM, ГИБКИЕ ПОДХОДЫ ПРИ РАЗРАБОТКЕ ПО	3
Теория	4
Темы.....	4
Что такое инструментальная система поддержки (управления) разработки ПО?	4
Состав TFS и Варианты установки (использования)	6
Команда проекта	9
Модель ЖЦ.....	10
MSF for CMMI	13
Scrum.....	21
Agile	23
Практика	24
Создать и настроить командный проект	24
Создание командного проекта	25
Настройка команды.....	27
Задание.....	30
02 ОБЕСПЕЧЕНИЕ КАЧЕСТВА КОДА, КОНТРОЛЬ ВЕРСИЙ	30
Теория	30
Создание проекта	30
Добавьте исходный код	31
Практика	33
Создание проекта	34
Настройка команды.....	37
Планирование работы.....	38
Добавьте исходный код	39
Задание.....	40
03 ТЕСТИРОВАНИЕ ПО И СБОРКА.....	40
Теория	40
Статический анализ кода	40
Система управления версиями	40

Ветвления и Слияния.....	41
Модульное тестирование	41
Автоматические сборки	41
Практика	42
Статический анализ кода	42
Система управления версиями.....	42
Ветвления и Слияния.....	43
Модульное тестирование	44
Автоматические сборки	44
Задание.....	48
04 АВТОМАТИЗИРОВАННОЕ ТЕСТИРОВАНИЕ	48
Теория	48
Тестирование приложения с помощью Microsoft Test Manager	48
Практика	48
Планирование ручных тестов с помощью Team Web Access	48
Задание.....	48
05 ДИАГНОСТИКА, ЭКСПЛУАТАЦИЯ, ОТЧЁТНОСТЬ.....	49
Теория	49
Практика	49
Задание.....	49

3.2. Инструментальная поддержка процесса разработки

- ▶ Версионирование проекта
 - Ветки, теги
 - Основные операции
 - Системы контроля версий
 - Поддержка нескольких версий ПО
- ▶ Управление дефектами и изменениями
 - Свойства дефекта
 - Жизненный цикл дефекта
 - Промышленные системы управления дефектами
- ▶ Сборка программных проектов
 - Основные задачи и проблемы
 - Управление зависимостями
 - Автоматизация сборки программных проектов
- ▶ Непрерывная интеграция
- ▶ Выпуск программного продукта
 - Дистрибутив
 - Альфа и бета-версии. Релиз программного продукта
 - Сопровождение программного продукта
- ▶ Конфигурационное управление
- ▶ Управление рисками



Ицыксон В.М. ТРПО/ОПИ © 2014

01 | ПРОЦЕССЫ ALM, ГИБКИЕ ПОДХОДЫ ПРИ РАЗРАБОТКЕ ПО

В данном модуле мы рассмотрим вызовы и проблемные области индустрии. Общие концепции Team Foundation Server, шаблоны процессов. Проектный вебсайт и командный обозреватель. Рабочие элементы и управление ими. Планирование и отслеживание работ. Формирование команды. Планирование итераций. Отслеживание работ и почтовые уведомления.

i
☰

01 | Процессы ALM, гибкие подходы...
01:27:00
100%

📺 Видео: Процессы ALM, гибкие подход...
✓

60 Минуты
3 Баллы

↓ Планирование и отслеживание проект...
✓

10 Минуты
1 Баллы

↓ Бесплатная 90-дневная пробная верси...
✓

10 Минуты
2 Баллы

☑ Выполнить самооценку
✓

7 Минуты
5 Баллы

Теория

Темы

Что такое инструментальная система поддержки (управления) разработки ПО?



...

Основное



Код. [Совместное использование и сборка кода с помощью подсистемы контроля версий Team Foundation \(TFVC\) или Git.](#)
[Разработка приложений с помощью TFVC](#) и [разработка приложений в репозитории Git.](#)



Работа. [Планирование проектов, отслеживание работы, сотрудничество в команде и составление отчетов о ходе работы.](#)

Создание списка невыполненных работ, работа в спринтах, построение диаграмм хода выполнения и совместная работа с использованием комнат рабочих групп. Построение раскадровки идей и получение отзывов о работающем программном обеспечении.



Сборка. [Настройка локального сервера сборки и определение процессов сборки.](#)
Также можно [настроить сборки непрерывной интеграции с использованием Visual Studio Online.](#)



Тестирование. [Тестирование приложения.](#)

Планирование тестов и отслеживание хода работы по каждой вехе. Выполнение ручных и автоматических тестов, включая тесты производительности и нагрузочные тесты.

Развертывание приложений в виртуальных средах для получения возможности осуществлять более сложную разработку и тестирование.

Дополнительное

После подготовки и запуска проекта можно использовать дополнительные средства для поддержки разработки, обеспечения качества кода и развертывания приложений



Разработка. [Моделирование приложений для достижения лучшего понимания и удовлетворения требований пользователей.](#)

Проектирование функциональности либо на базе существующих активов, либо с разработкой нового приложения. Использование архитектурных схем для визуализации кода и передачи важных сведений о программном обеспечении команды.



Диагностика. [Улучшение приложения с помощью средств диагностики.](#)

Запись, модульное тестирование, отладка, анализ и профилирование приложения.



Развёртывание. [Совершенствование процесса управления выпуском приложения.](#)

Развёртывание приложений в определённой среде для каждого отдельного этапа. Управление этапами в процессе с утверждением каждого этапа.



Контроль. [Обеспечение исправности работы приложения в реальном времени и изучение того, как оно используется.](#)

Применение средств контроля в веб-приложении в реальном времени для измерения его производительности и поиска отклонений от нормы и исключений, а также отслеживание его доступности из любой точки мира. Запись данных об использовании в журнал в развернутом веб-приложении или приложении для Windows Phone в целях улучшения настройки согласно потребностям пользователей.

Служба Application Insights в настоящее время доступна в предварительной версии.

Служебное



Администрирование. [Настройка серверов, добавление функциональности, управление доступом и разрешениями и обеспечение операций сервера.](#)

TFS предусматривает возможность применения как в небольших группах разработки, так и на крупных предприятиях.



Настройка/адаптация (рабочих процессов). [Настройка данных отслеживания, форм, рабочего процесса и других объектов для проекта.](#)

Для обеспечения соответствия рабочим процессам и организационной структуре компании можно настраивать многие элементы в TFS.

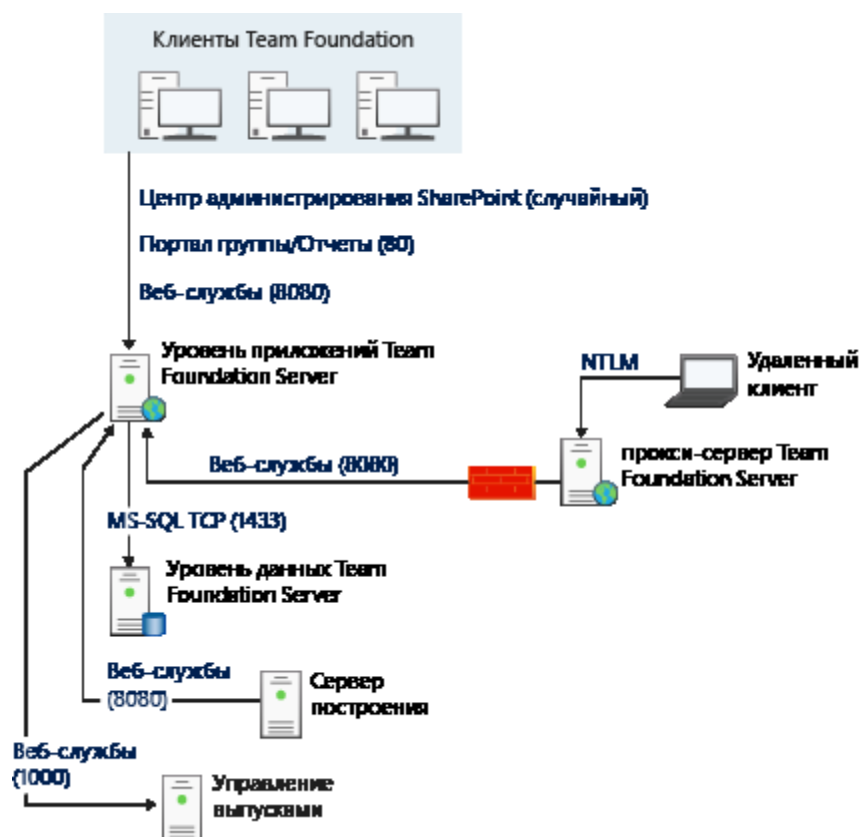
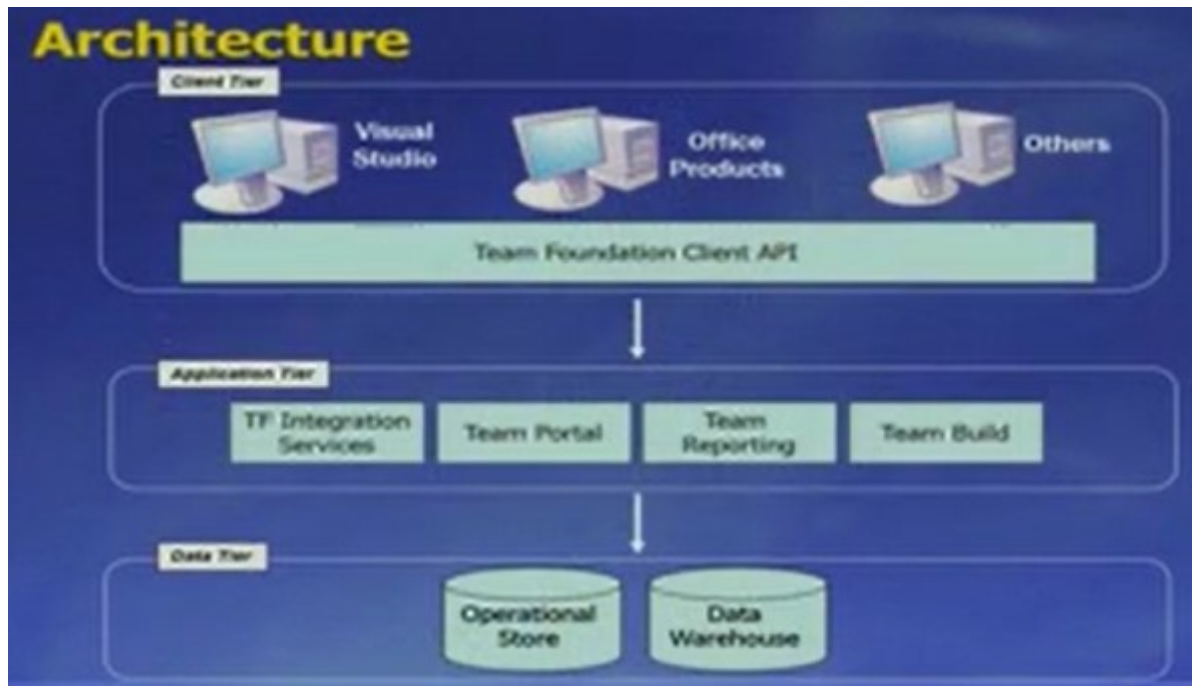


Настройка (функционирования). [Настройка TFS, создание командного проекта и добавление учетных записей участников команды.](#)

...

Состав TFS и Варианты установки (использования)

Состав



...

Клиенты Team Foundation

Дополнительные клиенты Team Foundation поддерживают возможностью подключения руководителей программ и других заинтересованных лиц к TFS. Team Explorer устанавливается с каждой версией Visual Studio; также можно установить Team Explorer из центра загрузки Майкрософт в качестве автономного клиента. Team Explorer поддерживает подключение к TFS и доступ к дополнительным клиентам Team Foundation. Дополнительные сведения см. в разделе [Выбор клиента Team Foundation для поддержки требуемых задач](#)

Основные клиенты	Клиенты определённых задач	Клиенты интеграции с Microsoft Office
Team Web Access (TWA)	Microsoft Test Manager	Excel
Team Explorer	Microsoft Feedback Client	Project
Team Explorer Everywhere	Ограниченный доступ TWA или просмотр только рабочего элемента	Project Professional

Термины и основные понятия

- **Уровень приложений, уровень данных и клиентский уровень:** логические уровни, составляющие Team Foundation Server. Все эти уровни могут быть развернуты на одном физическом компьютере или установлены на нескольких компьютерах. Для получения дополнительной информации см. [Архитектура Team Foundation Server](#).
- **Коллекция командного проекта:** основная организационная единица для всех данных в Team Foundation Server. Коллекции могут содержать один или несколько командных проектов. Для получения дополнительной информации см. [Управление коллекциями командных проектов](#).
- **Командный проект:** центральный элемент, с помощью которого координируются командные действия, необходимые для разработки определённой программной технологии или продукта. Командные проекты организованы в коллекции командных проектов. Для получения дополнительной информации см. [Отслеживание работ с помощью Visual Studio ALM и TFS](#).
- **Продукты SharePoint:** программное обеспечение, которое поддерживает порталы командных проектов и панели мониторинга. Вы можете включать одно или несколько веб-приложений SharePoint в рамках вашего развёртывания Team Foundation Server. Чтобы включить одно из этих приложений, необходимо установить и настроить расширения Team Foundation Server для Продукты SharePoint, а также необходимо настроить разрешения во всем развёртывании. Для получения дополнительной информации см. [Организация общего доступа к информации с помощью портала проекта](#).
- **SQL Server** и **SQL Server Reporting Services:** программное обеспечение, которое предоставляет платформу баз данных для хранения данных и платформу бизнес-аналитики для решений по интеграции данных, анализу и созданию отчетов. TFS хранит свои данные в базах данных SQL Server. Вы также можете дополнительно включить сервер, на котором выполняются SQL Server Reporting Services, и который автоматически создает отчеты для командных проектов. Для получения дополнительной информации см. [Управление отчетами, хранилищем данных и кубом служб Analysis Services TFS](#).

Варианты установки (использования)










- **Начало работы с использованием облачной службы**
https://dmitriykalachev.visualstudio.com/DefaultCollection/Test2015_Scrum/backlogs/Iteration/Sprint%201
- Первая установка — стандартная конфигурация
 базовую установку TFS для обмена кодом и улучшения совместной работы с клиентами и членами группы. Если у вас еще нет продукта TFS, вы можете установить его на своем сервере, выполнив действия из следующих подразделов данного раздела: Установка TFS, Установка службы сборок и Установка Team Explorer
- Первая установка — расширенная конфигурация

Слишком сложно


При использовании расширенной настройке в клиентской операционной системе доменные учетные записи не требуются. На сервере необходимо иметь хотя бы одну доменную учетную запись, которую можно использовать для всех перечисленных ниже удостоверений:

- учетной записи службы для Team Foundation Server;
- учетной записи читателя отчетов для создания HTML-отчетов;
- учетной записи службы для новых установок SharePoint Foundation.

В следующей таблице перечислены серверы, работу которых необходимо обеспечить в развёртывании Team Foundation, чтобы у команды был доступ к соответствующим функциям. Можно добавлять ресурсы после создания командного проекта. См. раздел [Вопрос. Как добавить ресурсы после создания командного проекта?](#).

Область приложения	Team Foundation Server	Team Foundation Build	Продукты SharePoint	Службы аналитики SQL Server	Службы отчетов SQL Server	Прокси-сервер Team Foundation Server	Visual Studio Lab Management
 Веб-доступ (1) <input checked="" type="checkbox"/>							
 Обозреватель управления исходным кодом (2) <input checked="" type="checkbox"/>							
 Рабочие элементы (3) <input checked="" type="checkbox"/>							
 Сборки (4) <input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>					
 Документы (5) <input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>				
 Отчеты Excel (6) <input checked="" type="checkbox"/>			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>			
 Отчеты (7) <input checked="" type="checkbox"/>				<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
 Поддержка удаленного сайта (8) <input checked="" type="checkbox"/>						Рекомендовано	
 Виртуальные среды (9) <input checked="" type="checkbox"/>							<input checked="" type="checkbox"/>

Примечания.

1.  Веб-доступ открывает **Team Web Access**, предоставляющий веб-интерфейс к TFS, дающий доступ к командным проектам, средствам планирования и отслеживания Agile, управлению версиями и сборкам. Некоторые функции требуют настройки. См. раздел [Изменение уровней доступа](#).
2. С помощью **Обозревателя управления исходным кодом** команда может управлять исходным кодом командного проекта. См. раздел [Использование управления версиями](#).
3. На странице **Рабочие элементы** команда может создавать рабочие элементы и запросы рабочих элементов для отслеживания, контроля и формирования отчётов о разработке продукта и его функций. Рабочий элемент — это запись в базе данных, содержащая определение, присвоение, приоритет и состояние работы. Команда может создавать только рабочие элементы типов, определённых в шаблоне процесса, использованном при создании командного проекта, или типов, добавленных к командный проект после его создания. См. раздел [Создание списка невыполненной работы](#).
4. На странице **Сборки** перечислены определения сборок, определённые для вашего командного проекта. Эта страница отображается, только если в рамках развёртывания установлена и настроена служба Team Foundation Build. Для доступа к Служба построения Team Foundation требуется, чтобы коллекция проектов была настроена на использование контроллера построений. Для каждой коллекции командных проектов выделяется отдельный контроллер построений. Контроллер принимает запросы на построение от всех командных проектов в указанной коллекции. Team Foundation Build позволяет команде регулярно создавать сборки продуктов и управлять ими. Например, команда может выполнять ежедневные построения и отправлять их на общий сервер.

Team Foundation Build также дает возможность создавать отчеты о состоянии и качестве каждого построения. См. раздел [Сборка приложения](#).

5. Страница **Документы** отображается, только если командный проект настроен с [Продукты SharePoint](#). После создания командного проекта можно настроить сайт SharePoint или другое расположение портала проекта в Интернете. См. раздел [Совместное использование информации с помощью портала проекта](#).
6. Отчеты Microsoft Excel добавляются в папку Документы при настройке командного проекта с сайтом SharePoint. Эти отчеты позволяют отслеживать скорость выполнения командного проекта, объем невыполненной работы с ошибками, качество программного обеспечения, ход выполнения тестов и другие показатели. Многие из этих отчетов отображаются на панелях мониторинга проекта. Помимо зависимости Продукты SharePoint, отчеты Excel зависят от того, настроены ли для коллекции командных проектов, размещающей командный проект, обе службы аналитики SQL Server. Обзор отчетов Excel по умолчанию см. в разделе [отчеты Excel](#) или [Отчеты Excel \(CMMI\)](#). Если у командного проекта нет сайта SharePoint, можно использовать Excel для создания отчетов о состоянии и тенденциях. См. раздел [Создание отчетов Excel при помощи запроса рабочего элемента](#).
7. Страница Отчеты открывается, только если коллекция проектов, размещающая командный проект, настроена со службами отчетов SQL Server и службами аналитики SQL Server. Эта страница предоставляет доступ к диспетчеру отчетов, пользовательским отчетам и отчетам по умолчанию, добавленным на сервер, где размещены службы отчетов SQL Server. Каждый из шаблонов процесса по умолчанию, предоставленный Team Foundation Server, включает несколько отчетов по умолчанию. Общие сведения об отчетах по умолчанию см. в разделе [Отчеты \(SQL Server Reporting Services\)](#).
8. Если некоторые участники команды удалены от основного местоположения системы управления версиями, для поддержки их работы может потребоваться установить и настроить прокси-сервер Team Foundation Server. Прокси-сервер Team Foundation Server управляет кэшем загруженных файлов системы управления версиями в расположении распределенной команды, что существенно снижает требования к пропускной способности при работе в глобальной сети. Благодаря настройке клиентов на использование прокси-сервера Team Foundation Server, управление файлами становится прозрачным для пользователя. Любой обмен метаданными и загрузка файлов на сервер продолжает взаимодействовать непосредственно с TFS. См. раздел [Практическое руководство. Установка прокси-сервера Team Foundation и настройка удаленного сайта](#).
9. С помощью Visual Studio Lab Management команда может создавать, назначать и отслеживать виртуальные среды для поддержки разработки, развертывания и тестирования приложений. С помощью Microsoft Test Manager можно тестировать приложения в виртуальных средах. Для доступа к виртуальным средам из Test Manager требуется, чтобы сервер, на котором располагается командный проект, был настроен на взаимодействие с Lab Management. См. раздел [Настройка Lab Management для сред SCVMM](#).

Команда проекта

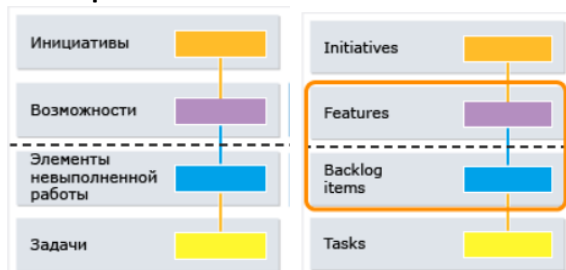
См. «Обзор Microsoft Solutions Framework (MSF)» <https://msdn.microsoft.com/ru-ru/library/jj161047.aspx>

Модель группы MSF разделяет типичные операции для реализации решения и ответственности в семь групп. Эти группы взаимозависимы и многоплановы. Как указано в таблице ниже, чтобы использовать сбалансированный подход, эти роли могут предоставить уникальный взгляд на то, что необходимо для проекта, что необходимо отстаивать и какие цели должны быть связаны с реализацией решения. Эти роли можно объединять для небольших групп и расширять для больших групп.

Эти роли не подразумевают какую-либо организационную диаграмму или набор должностей, так как они сильно отличаются в разных организациях и группах. Чаще всего эти роли распределяются среди различных групп в ИТ-организации и, иногда, в сообществе бизнес-пользователей, а также среди внешних консультантов и партнеров.

Роль	Цели	Функциональные области
Управление продуктами	Обеспечение бизнес-ценности решения Определение решения в рамках ограничений проекта Обеспечение выполнения потребностей и ожиданий клиентов	Маркетинговые и корпоративные коммуникации Бизнес-аналитика Планирование продукта Управление проектом Управление программой
Управление программой	Реализация решения в рамках ограничений проекта Реализация средств, с помощью которых выполняются потребности и ожидания заказчика	Управление ресурсами Обеспечение выполнения процессов Управление качеством проекта Операции проекта
Архитектура	Разработка решения для достижения бизнес-целей в рамках ограничений проекта	Архитектура решения Техническая архитектура
Разработка	Построение решения в соответствии со спецификациями	Разработка решения Технологические консультации Специальные возможности Интернационализация
Взаимодействие с пользователем	Оптимизация удобства использования решения Улучшение готовности пользователей к работе и ее эффективности Обеспечение выполнения потребностей и ожиданий пользователей	Взаимодействие со службой технической поддержки Обучение Удобство использования Проектирование пользовательского интерфейса
Тест	Утверждение выпуска решения после проверки того, что все его аспекты соответствуют заданным уровням качества или превосходят их	Регрессионное тестирование Функциональное тестирование Тестирование удобства использования Системное тестирование
Выпуск и эксплуатация	Простое развёртывание и переход к эксплуатации Обеспечение выполнения потребностей и ожиданий ИТ- и бизнес-подразделений	Управление выпусками Инфраструктура доставки Операции Управление сборками Администрирование

Модель ЖЦ



- Оранжевый – инициатива
- Сиреневый – Возможность
- Синий – Элемент невыполненной работы
- Жёлтый – задача
- Красный – Ошибка
- Зелёный – ??? Проблема

<p>Microsoft Visual Studio Scrum 2013</p> <p>Выберите Visual Studio Scrum, если команда управляет ошибками вместе с элементами невыполненной работы по продукту во время планирования спринта.</p> <p>Шаблон Scrum предназначен для поддержки методологии Scrum в соответствии с организацией. Этот шаблон процесса отслеживает ошибки на том же уровне, что и элементы невыполненной работы, а оценки трудозатрат отслеживаются с помощью поля "Трудозатраты".</p> <p>Система автоматически обновляет поле "Оставшиеся трудозатраты", когда состояние задачи устанавливается в значение "Готово".</p>	
<p>MSF for Agile Software Development 2013</p> <p>Выберите Agile, если в вашей организации ошибки рассматриваются отдельно от невыполненной работы по продукту, а рабочие элементы разрешаются, прежде чем их закрывают. Кроме того, выберите Agile, если команда выделяет время на ошибки в каждом спринте.</p> <p>Шаблон Agile разработан для поддержки методологии гибкой разработки для команд, которые не хотят быть ограничены методологией Scrum. Он поддерживает оценку пользовательских историй с помощью баллов истории. Задачи содержат поля для отслеживания полей "Исходная предварительная оценка", "Оставшаяся работа" и "Завершенная работа". Ошибки на страницах невыполненной работы не отслеживаются. Подробнее о методологиях Agile см. на странице http://www.agilealliance.org/.</p>	
<p>MSF for CMMI Process Improvement 2013</p> <p>Выберите CMMI, если в организации ошибки рассматриваются отдельно от невыполненной работы по продукту, рабочие элементы разрешаются, прежде чем их закрывают, а также формально отслеживаются изменения в требованиях.</p> <p>Шаблон CMMI предназначен для поддержки формальных процессов управления изменениями. Этот шаблон поддерживает оценку требований с помощью поля "Размер". Задачи содержат поля для отслеживания полей "Исходная предварительная оценка", "Оставшаяся работа" и "Завершенная работа". Ошибки на страницах невыполненной работы не отслеживаются.</p> <p>Подробнее о процессах CMMI см. здесь.</p>	

Состояния

...см. ...???

Вопросы, которые требуется задать команде (организация взаимодействия) для выбора модели

Для эффективного отслеживания работы участники команды должны согласовать, как они будут использовать типы рабочих элементов и инструменты. Ниже приведено несколько вопросов, на которые должна ответить команда.

Вопрос	Выбор команды
Как ваша команда отслеживает работу?	<p>Если члены команды в основном отслеживают ход выполнения работ путем обновления состояния элементов невыполненной работы, они могут использовать канбан-доску. Команда также может настраивать доску канбана для отслеживания хода выполнения в различных дорожках.</p> <p>Если команда разбивает элементы невыполненной работы на задачи для каждого спринта и оценивает оставшуюся работу, можно использовать доску задач спринта. Хотя оставшаяся работа обычно измеряется в часах, можно использовать совершенно любую единицу времени при условии, что она согласована командой. Путем вычисления и обновления оставшихся трудозатрат команда может отслеживать ход выполнения по диаграмме сгорания, предоставляемой на каждом спринте.</p>
Ваша команда отслеживает производительность по отдельным членам или по активности?	<p>Если команда отслеживает оставшуюся работу через задачи, она может оценить ресурсы для спринта для отдельных участников команды или для различных действий команды, таких как проектирование, тестирование и разработка.</p>
Как ваша команда группирует работу?	<p>Можно группировать работу несколькими способами. Элементы, создаваемые на странице невыполненной работы, автоматически назначаются пути к области команды. Элементы, назначенные спринту, назначены пути итерации спринта. Кроме того, можно назначать теги рабочим элементам, чтобы фильтровать невыполненную работу или список результатов запроса.</p>
Ваша команда использует скорость и прогнозирование?	<p>Для поддержки прогнозирования команда может использовать поля "Трудозатраты" (Scrum), "Баллы истории" (Agile) или "Размер" (CMMI), чтобы определить, сколько элементов может быть выполнено в течение спринта.</p>

Вопрос

Выбор команды

Как ваша команда обменивается информацией?

Ваша команда поддерживает свертку хода выполнения по нескольким командам?

Кроме того, диаграмма скорости покажет успех команды от спринта к спринту. Члены команды могут присоединять файлы к рабочим элементам, возвращать файлы в исходный код или совместно работать с использованием портала командного проекта. После [настройки портала проектов](#) команда имеет доступ ко всем функциям сайта SharePoint, включая библиотеки документов, wiki-страницы, блог и календарь событий.

С невыполненными работами портфеля можно быстро [просмотреть свертку выполняемой работы по нескольким командам](#). Если член команды работает в нескольких командах, он может [распределять свою производительность соответствующим образом между всеми командами](#).

Что же такое Agile?

- Мы находим лучшие подходы к разработке ПО, непосредственно участвуя в процессе разработки и помогая другим. В процессе работы мы пришли к тому, что для нас важнее:
- Люди и их взаимодействие, чем процессы и средства
- Работающее ПО, чем исчерпывающая документация
- Сотрудничество с заказчиком, чем обсуждение условий контракта
- Реагирование на изменения, чем следование плану

Планирование проекта

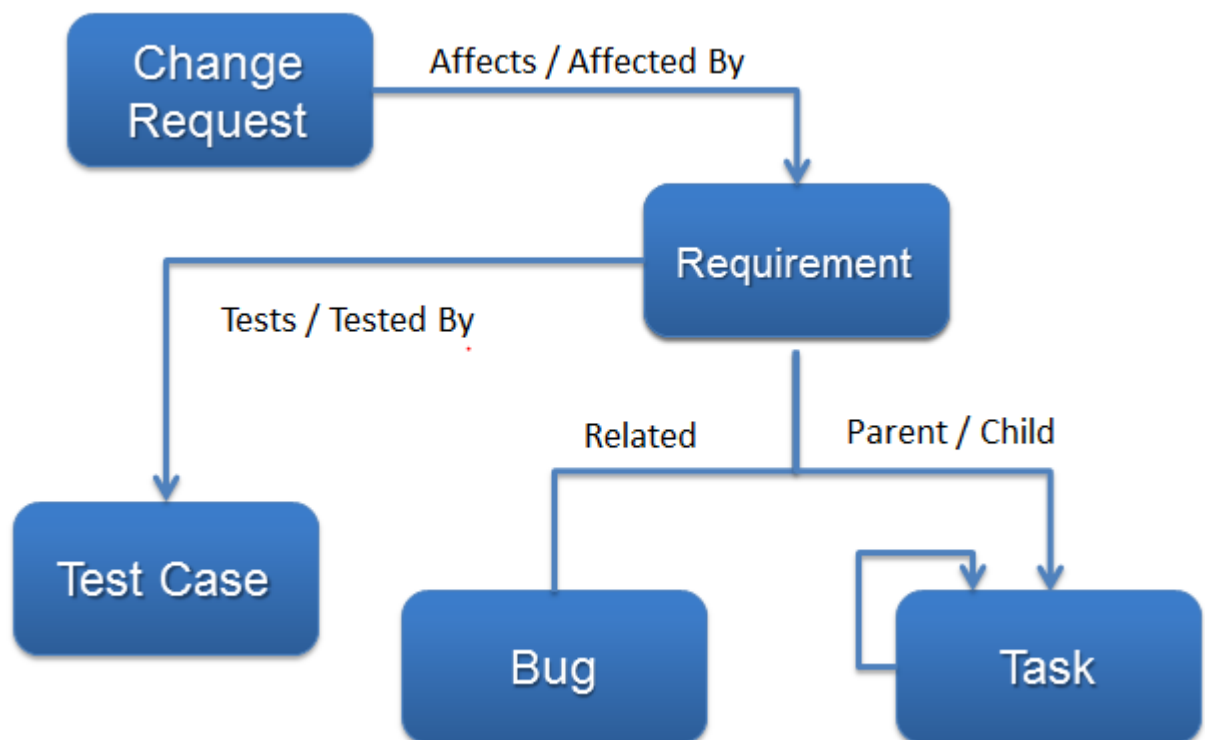
- Product Backlog (бэклог)
 - Перечень требований (пользовательских историй)
 - Описывает что хочет заказчик
 - Ответственный – Владелец Продукта (Product Owner)
 - Приоритеты по ценности для бизнеса
 - Каждое требование (пользовательская история) написано с точки зрения пользователя



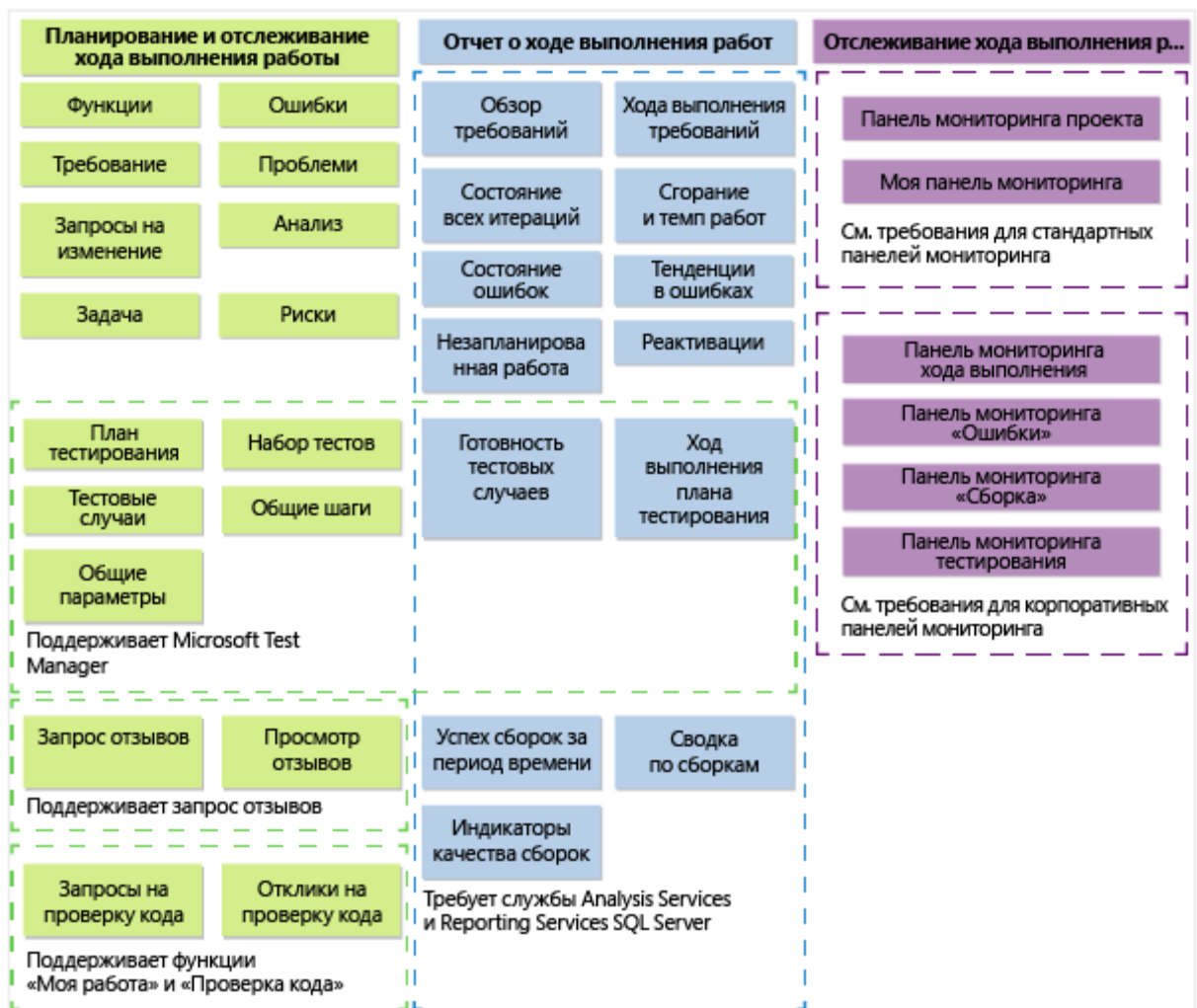
Как планируется спринт?

- Владелец продукта вместе с командой добавляет истории в спринт
- Каждая история декомпозируется до задач
- Каждый член команды делает оценку сроков на уровне задач и обязательства по закрытию истории к дате

MSF for CMMI



[https://msdn.microsoft.com/ru-ru/library/dd997574\(v=vs.120\).aspx](https://msdn.microsoft.com/ru-ru/library/dd997574(v=vs.120).aspx)



Планирование и отслеживание работы с использованием типов рабочих элементов CMMI

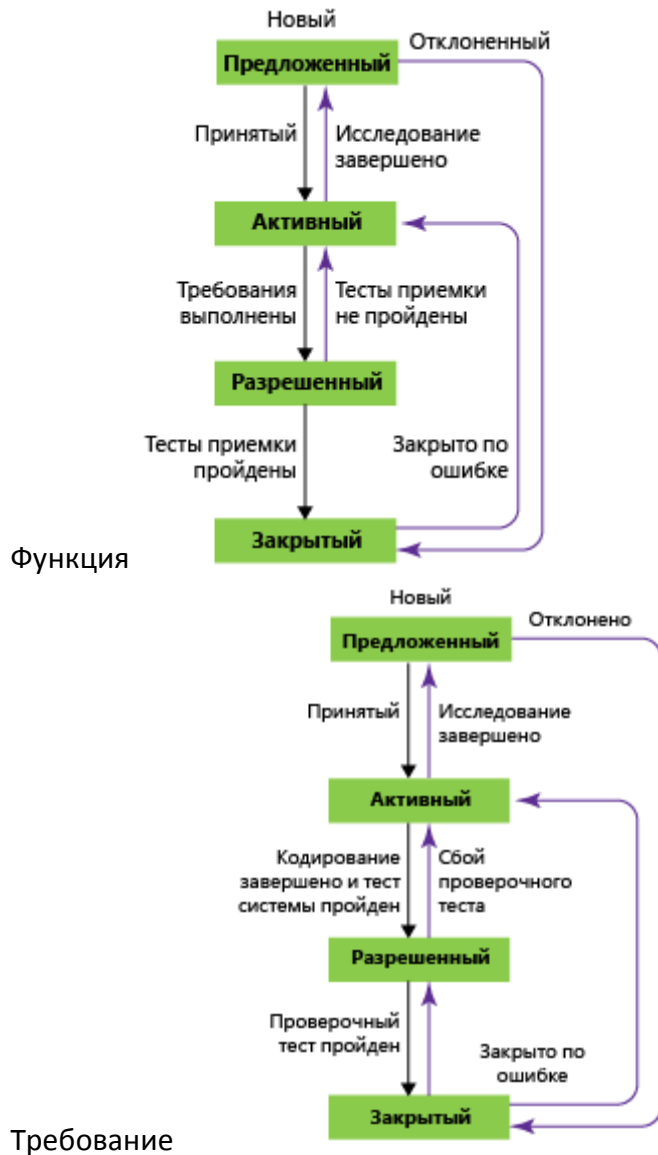
Команды планируют свой проект путем записи функций и требований. Если команды работают в спринтах, они определяют задачи и связывают их с требованиями. Для получения представления о свертке требований в разных командах руководители программ связывают требования с функцией. Блокирующие проблемы отслеживаются с помощью проблем. Сведения об использовании этих типов рабочих элементов см. в разделе [Типы и рабочий процесс рабочего элемента шаблона процесса CMMI](#)



Рабочий элемент — это

Рабочий элемент — это запись в базе данных TFS, содержащая определение, присвоение, приоритет и состояние работы. Типы рабочих элементов определяют шаблоны полей, рабочих процессов и форму для каждого типа. Рабочие элементы можно связать друг с другом, чтобы поддерживать отслеживание зависимостей, сведение работ и отчеты.

Данные диаграммы показывают основные состояния прогрессии и регрессии «Функция», «Требование», «Ошибка» и «Задача». Для настройки рабочего процесса перейдите [сюда](#).



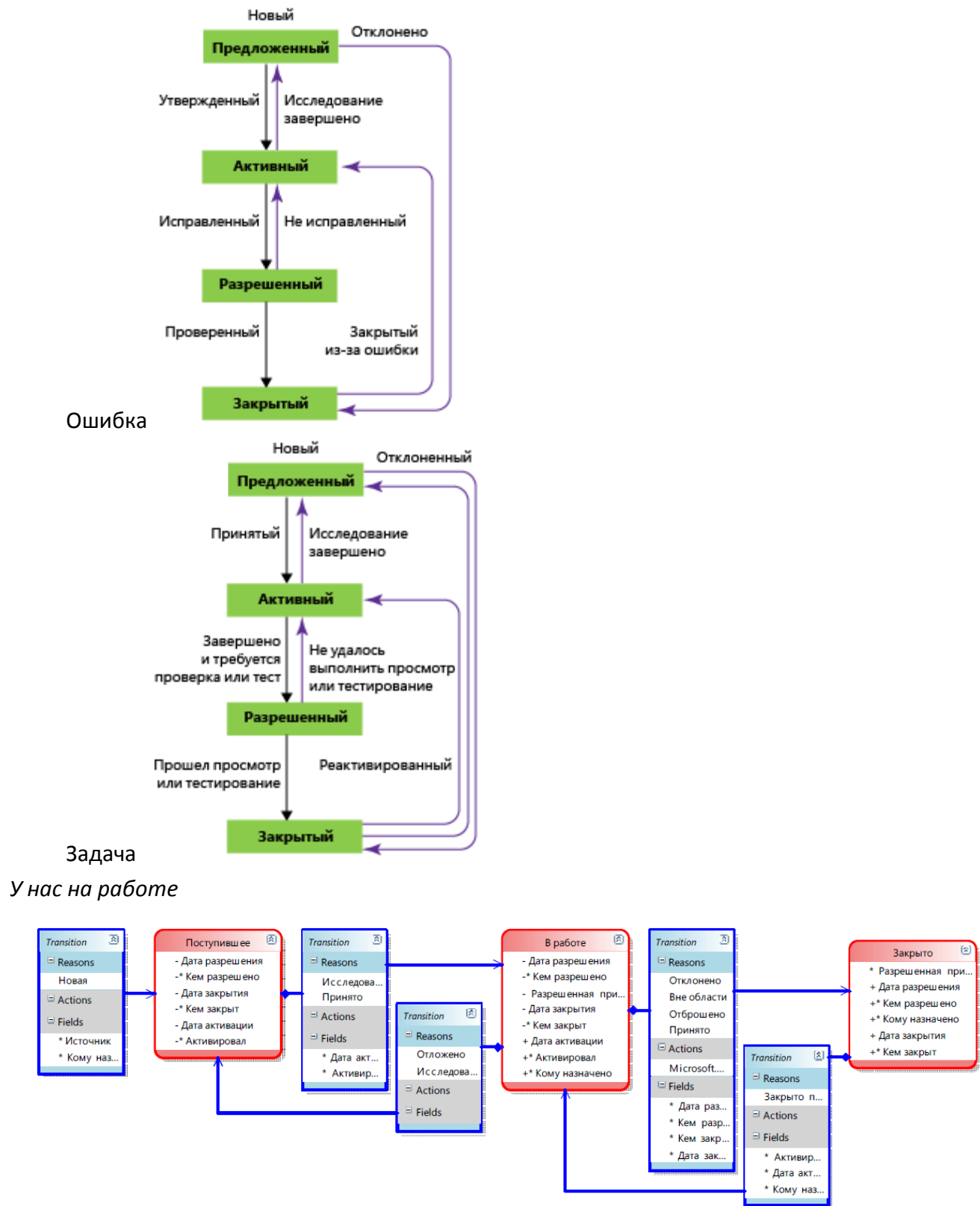


Рис. 1 Жизненный цикл рабочего элемента «Запрос на изменение»

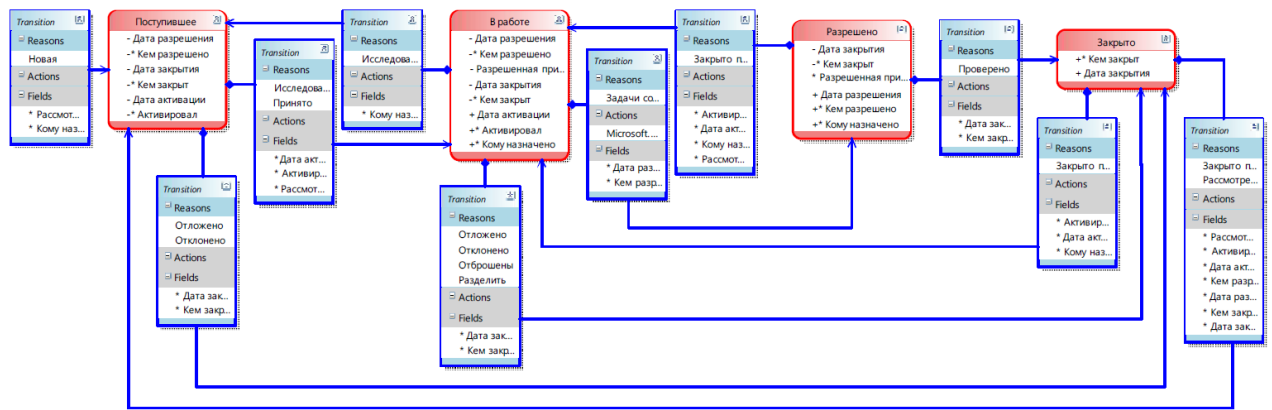


Рис. 2 Жизненный цикл рабочего элемента «Требование»

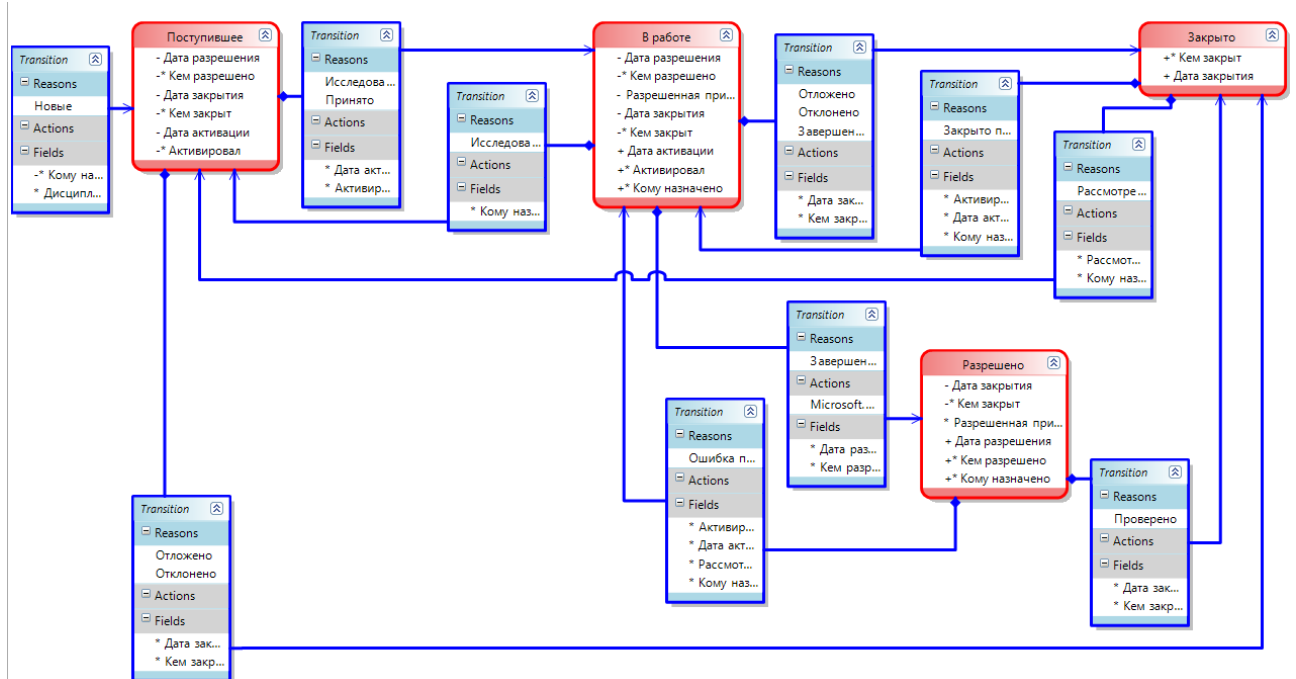


Рис. 3 Жизненный цикл рабочего элемента «Задача»

Жизненный цикл рабочего элемента «Ошибка»

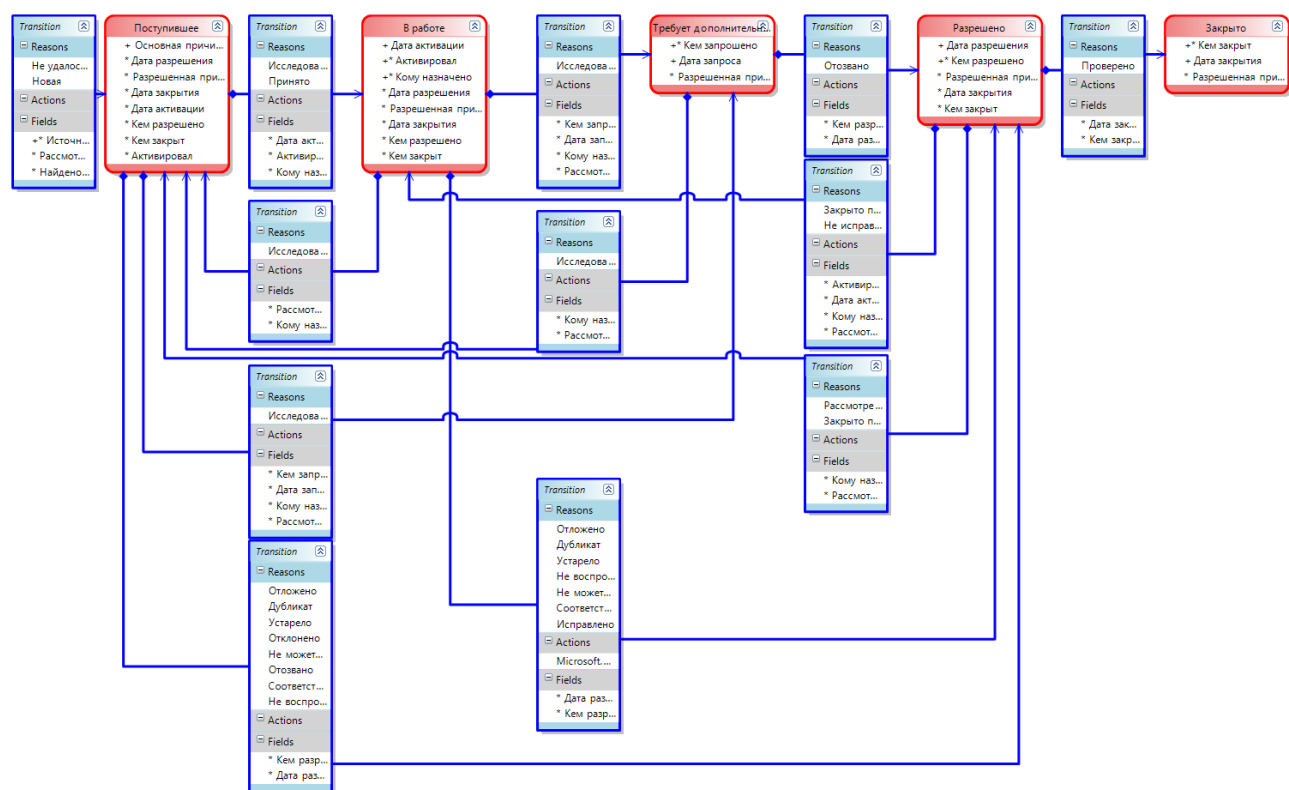


Рис. 4 Жизненный цикл рабочего элемента «элемент»

Типовая последовательность

№	Действие	Исполнители	Исходное состояние	Конечное состояние	Комментарий
1	Создание ошибки, причина: Новая	Тестировщик / Сопроводитель		Поступившее	Должны быть заполнены обязательные атрибуты.
2	Назначение исполнителя, причина Принять или Исследовать	Разработчик / Проектировщик	Поступившее	В работе	Должен быть назначен исполнитель, как правило, разработчик
3	Устранение ошибки, причина Исправлено	Разработчик	В работе	Разрешено	Ошибка исправлена
4	Проверка исправления, причина Проверено	Тестировщик	Разрешено	Закрото	Если ошибка была исправлена корректно
5	Проверка исправления, причина Не исправлено	Тестировщик	Разрешено	В работе	Если ошибка не была исправлена корректно
6	Включение в версию	Интегратор	Закрото		Фиксируется включение исправления в очередную версию

Запрос дополнительной информации

№	Действие	Исполнители	Исходное состояние	Конечное состояние	Комментарий
1	Создание ошибки, причина: Новая	Тестировщик / Сопроводитель		Поступившее	Должны быть заполнены обязательные атрибуты.

№	Действие	Исполнители	Исходное состояние	Конечное состояние	Комментарий
3	Запрос дополнительной информации, причина: Исследовать	Разработчик / Аналитик / Проектировщик	Поступившее	Требуется дополнительная информация	Должна быть представлена требуемая информация или ошибка отозвана.
1	Создание ошибки, причина: Новая	Тестировщик / Сопроводитель		Поступившее	Должны быть заполнены обязательные атрибуты.
2	Назначение исполнителя, причина Принять или Исследовать	Разработчик / Проектировщик	Поступившее	В работе	Должен быть назначен исполнитель, как правило, разработчик
3	Запрос дополнительной информации, причина: Исследовать	Разработчик / Проектировщик	В работе	Требуется дополнительная информация	Должна быть представлена требуемая информация или ошибка отозвана.
4	Требуемая информация представлена, причина: Исследование завершено	Автор ошибки	Требуется дополнительная информация	Поступившее	Требуемая информация представлена, ошибка д.б. рассмотрена повторно.
4	Требуемая информация не представлена, причина: Отозвано	Автор ошибки / Менеджер (по истечению нормативного срока)	Требуется дополнительная информация	Разрешено	Требуемая информация не представлена, или ошибка потеряла актуальность.

Отклонение

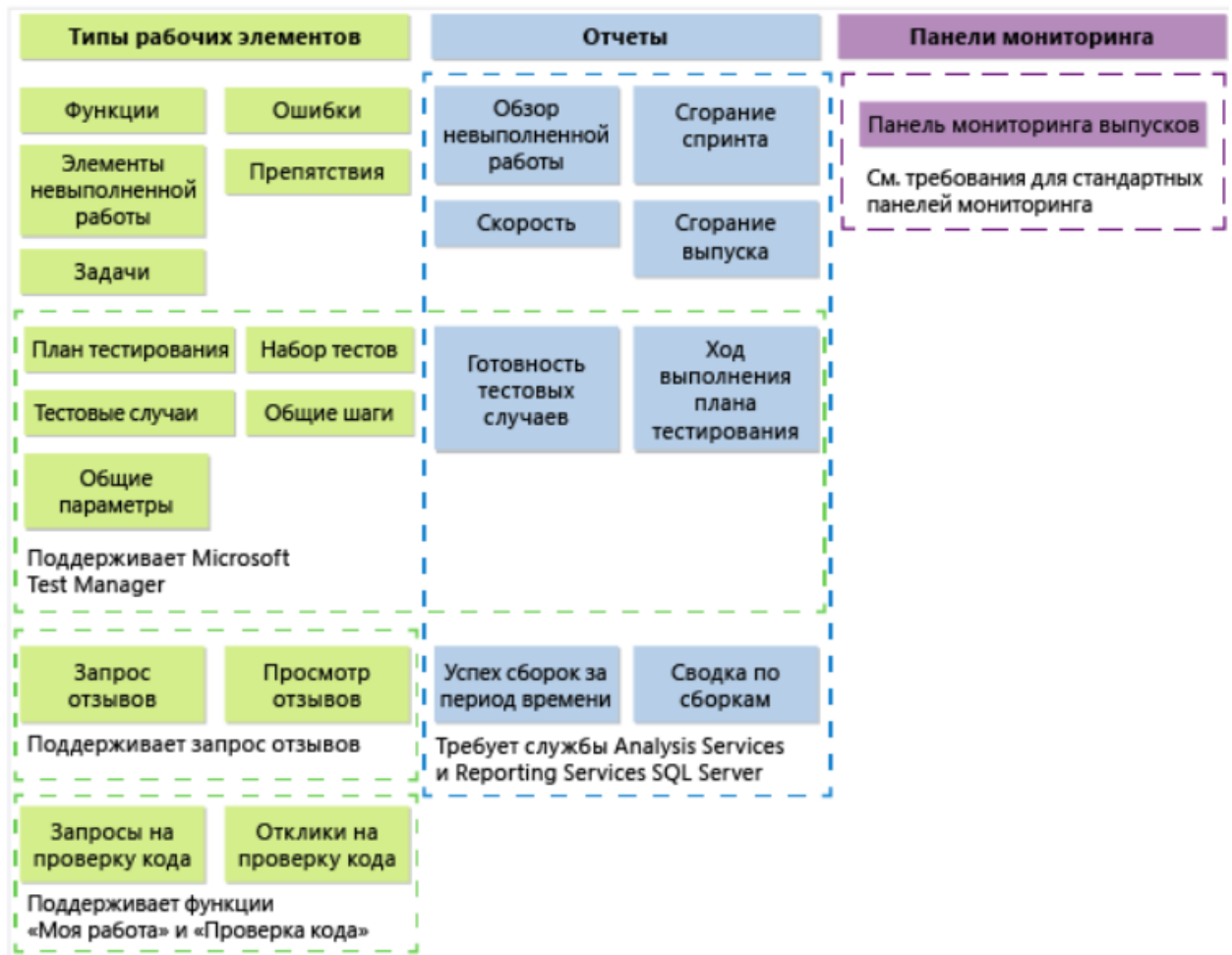
№	Действие	Исполнители	Исходное состояние	Конечное состояние	Комментарий
1	Создание ошибки, причина: Новая	Тестировщик / Сопроводитель		Поступившее	Должны быть заполнены обязательные атрибуты.
3	Отклонение ошибки, причина: Не воспроизводится Дубликат Устарело Отклонено Соответствует проекту Не может быть исправлено	Разработчик / Аналитик / Проектировщик	Поступившее	Разрешено	В случае не подтверждения ошибки или нецелесообразности ее исправления.
1	Создание ошибки, причина: Новая	Тестировщик / Сопроводитель		Поступившее	Должны быть заполнены обязательные атрибуты.
2	Назначение исполнителя, причина Принять или Исследовать	Разработчик / Проектировщик	Поступившее	В работе	Должен быть назначен исполнитель, как правило, разработчик

№	Действие	Исполнители	Исходное состояние	Конечное состояние	Комментарий
3	Отклонение ошибки, причина: Не воспроизводится Дубликат Устарело Отклонено Соответствует проекту Не может быть исправлено	Разработчик / Проектировщик	Поступившее	Разрешено	В случае не подтверждения ошибки или нецелесообразности ее исправления.
4	Возможный возврат в работу, причина: Закрывается по ошибке	Тестировщик / Аналитик / Сопроводитель	Разрешено	Поступившее	Если ошибка была не исправлена а закрыта по другой причине, но тестировщик считает это неверным.

Приостановка рассмотрения

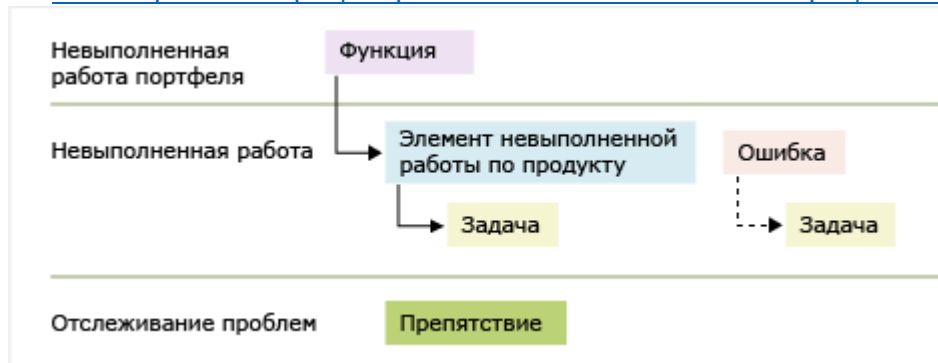
№	Действие	Исполнители	Исходное состояние	Конечное состояние	Комментарий
1	Создание ошибки, причина: Новая	Тестировщик / Сопроводитель		Поступившее	Должны быть заполнены обязательные атрибуты.
3	Отложит ошибку на более поздний срок, причина: Отложено	Разработчик / Аналитик / Проектировщик	Поступившее	Разрешено	Если исправление в текущей итерации невозможно или нецелесообразно.
1	Создание ошибки, причина: Новая	Тестировщик / Сопроводитель		Поступившее	Должны быть заполнены обязательные атрибуты.
2	Назначение исполнителя, причина Принять или Исследовать	Разработчик / Проектировщик	Поступившее	В работе	Должен быть назначен исполнитель, как правило, разработчик
3	Отложит ошибку на более поздний срок, причина: Отложено	Разработчик / Аналитик / Проектировщик	В работе	Разрешено	Если исправление в текущей итерации невозможно или нецелесообразно.
4	Вернуть ошибку к рассмотрению, причина: Рассмотреть	Аналитик / Проектировщик / Сопроводитель	Разрешено	Поступившее	Команда определилась с итераций, в которую исправление ошибки д.б. включено.

Scrum



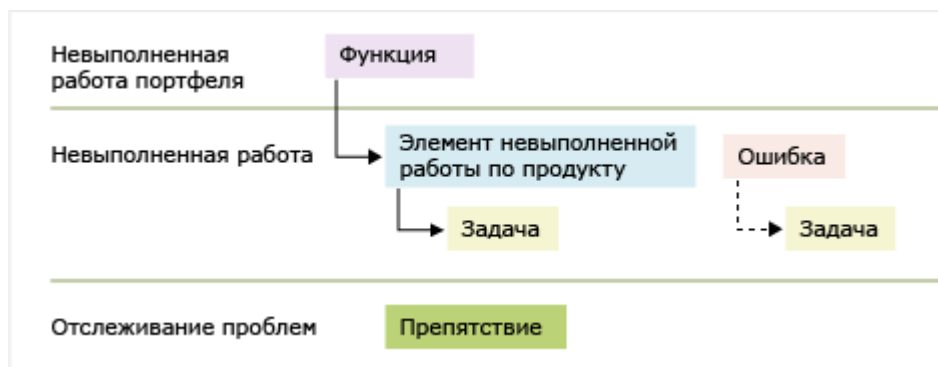
Планирование и отслеживание работы с использованием типов рабочих элементов

Команды планируют свой проект путем формирования функций и требований как элементов невыполненной работы по продукту (PBI). Они отслеживают ошибки, работу и блокирующие проблемы с помощью рабочих элементов типов "ошибка", "задача" и "препятствие". Функции поддерживают первый уровень управления портфелем для просмотра свертки элементов PBI по командам. Подробные сведения об использовании этих WIT см. в разделе [Типы и рабочий процесс рабочего элемента шаблона процесса Scrum](#).



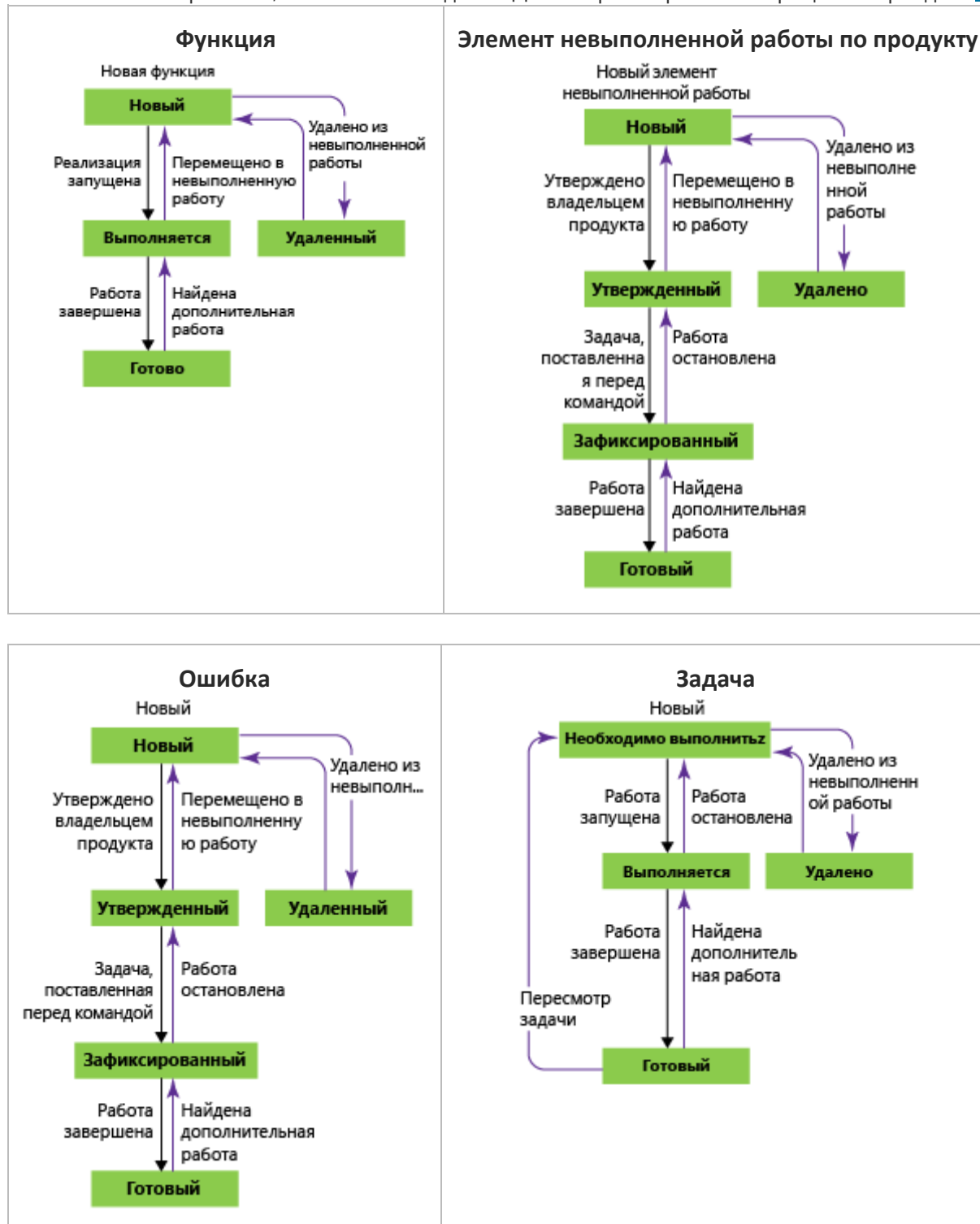
Типы и рабочий процесс рабочего элемента шаблона процесса Scrum

Для планирования проекта по разработке программного обеспечения и отслеживания дефектов ПО с помощью Scrum команды используют типы рабочих элементов "**Элемент невыполненной работы**" (PBI - **product backlog item**) и "Ошибка". Для лучшего понимания портфеля функций, сценариев или взаимодействия с пользователем владельцы продуктов и руководители программ могут сопоставлять PBI и ошибки с функциями. Если команды работают в спринтах, они определяют задачи, которые автоматически связываются с PBI и ошибками.



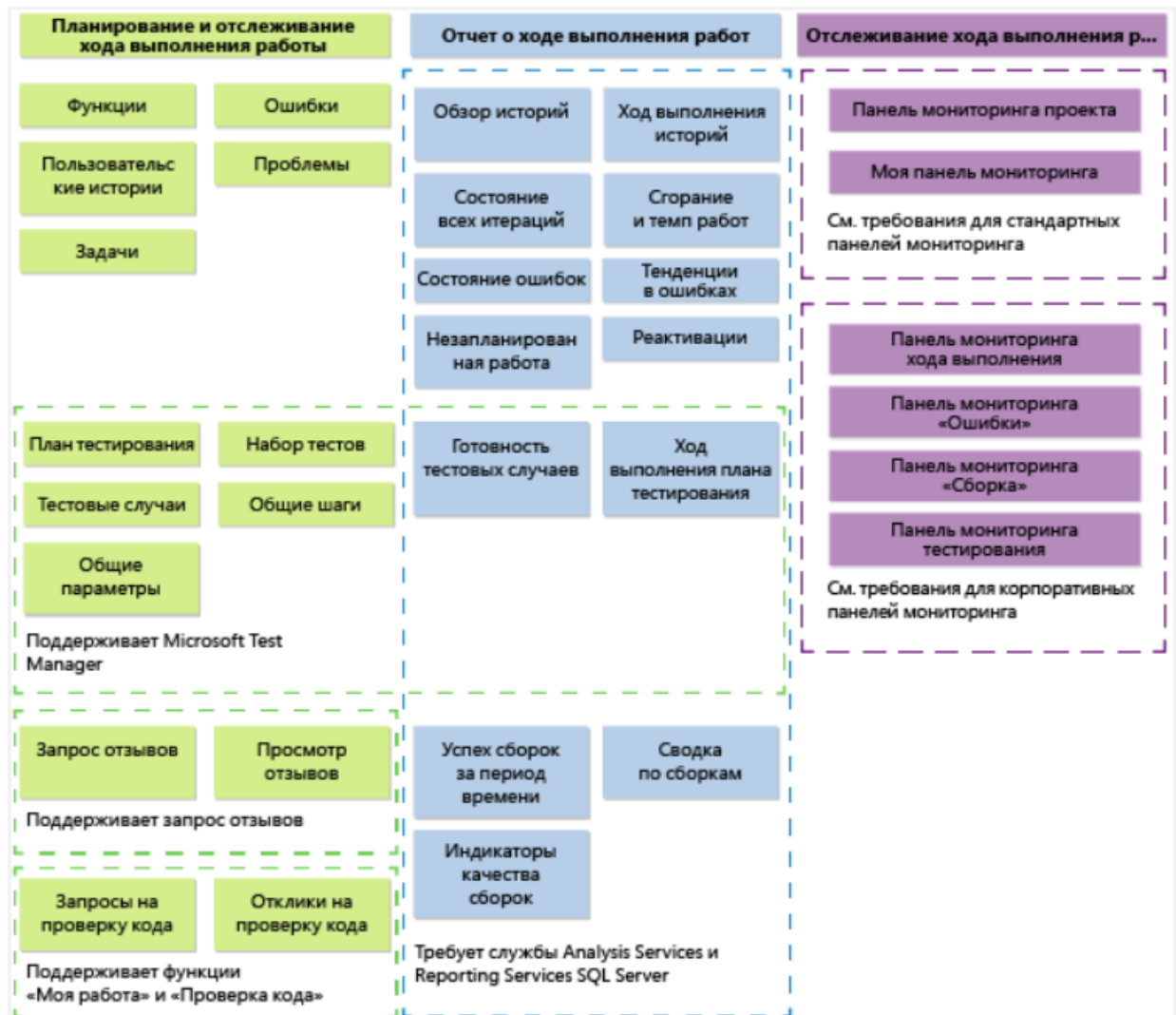
Рабочие элементы

Данные диаграммы показывают основные состояния прогрессии и регрессии «Функция», «Элемент невыполненной работы», «Ошибка» и «Задача». Для настройки рабочего процесса перейдите [сюда](#).



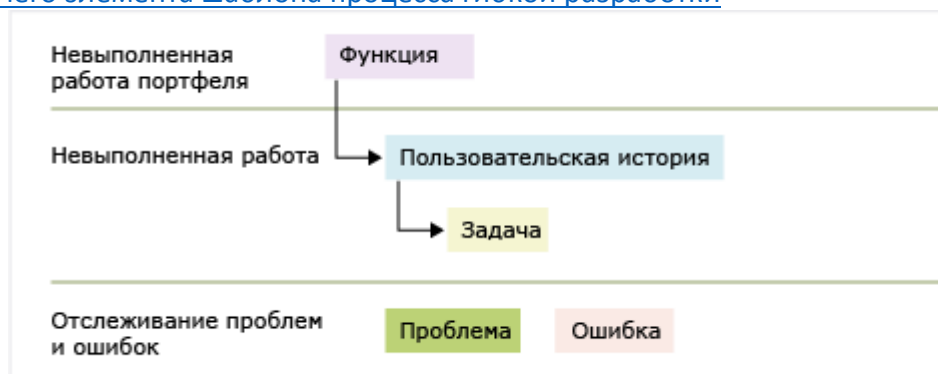
Agile

MSF for Agile Software Development для Visual Studio ALM <https://msdn.microsoft.com/ru-ru/library/dd380647.aspx>



Планирование и отслеживание работы с использованием типов рабочих элементов

Команды планируют свой проект путем фиксации функций и требований в виде пользовательских историй. Они отслеживают ошибки, работу и блокирующие проблемы с помощью ошибки, задачи и WIT проблем. Для поддержки управления портфелем команды создают функции для просмотра свертки пользовательских историй в различных командах. Подробные сведения об использовании этих WIT см. в разделе [Типы и рабочий процесс рабочего элемента шаблона процесса гибкой разработки](#)



Рабочий элемент — это запись в базе данных TFS, содержащая определение, присвоение, приоритет и состояние работы. Типы рабочих элементов определяют шаблоны

полей, рабочих процессов и форму для каждого типа. Рабочие элементы можно связать друг с другом, чтобы поддерживать отслеживание зависимостей, сведение работ и отчеты.

Функция – ...???

Пользовательские истории определяют приложения, требования и элементы, которые командам необходимо создать. Определяют и ранжируют пользовательские истории обычно владельцы продукта. Затем команда оценивает трудозатраты и объем работ, необходимые для поставки элементов с наивысшим приоритетом.

Когда управление работой команды организовано в виде спринтов, можно использовать [страницу невыполненных работ спринта](#) для разбиения работы, которую необходимо выполнить, на определенные **задачи**.

...

Тестовые случаи. Из Test Manager или из TWA можно [создавать тестовые случаи, которые автоматически связываются с пользовательской историей или ошибкой](#).

...

Проблема – ???

Отслеживание дефектов – **ошибки**

...

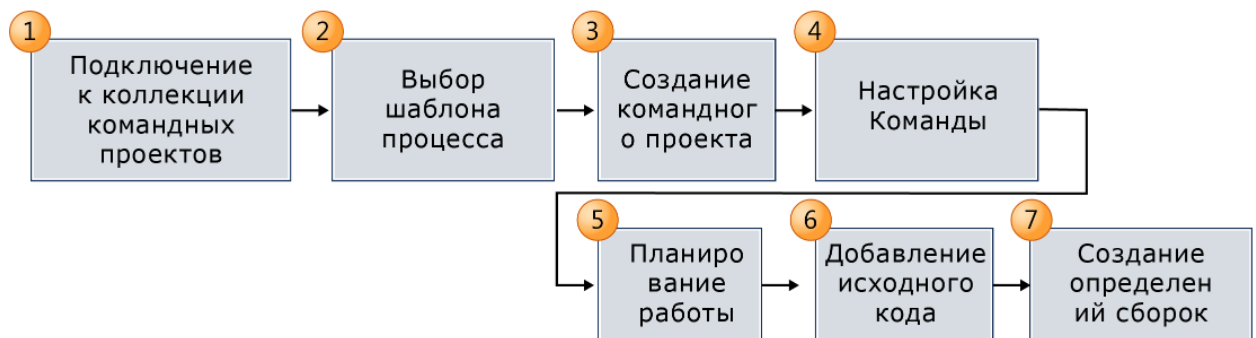
Практика

Создать и настроить командный проект

Планирование и начало работы с новым командным проектом

<https://msdn.microsoft.com/ru-ru/library/ee748449.aspx>

Добавление командного проекта в существующую установку



1. Подключение к коллекции командных проектов

Теория

В Visual Studio откройте Team Explorer и подключитесь к коллекции командных проектов, в которой нужно создать командный проект. Командные проекты, находящиеся в одной коллекции командных проектов, используют аналогичные ресурсы. Для получения дополнительной информации см. [Подключение к командным проектам на сервере Team Foundation Server](#).

2. Выбор шаблона ...

Scrum, ранее в Теории

Выбор шаблона процесса Шаблон процесса определяет типы рабочих элементов и другие артефакты, которые команда может использовать для отслеживания работы и управления ею. Команда может создавать только рабочие элементы типов, определённых в шаблоне процесса, использованном при создании командного проекта, или типов, добавленных к командный проект после его создания.

3. Создание командного проекта

Практика

Определение структуры ветвей для исходного кода. При создании командного проекта указывается пустая папка для исходного кода или создаётся новая ветвь под управлением систему управления версиями

4. Настройте команды

Практика

Команда по умолчанию соответствует командному проекту. Можно создать дополнительные команды для организации, отслеживания и упрощения работы групп людей в одной области продукта. Для командного проекта можно создать любое количество команд.

Уровень доступа участников команды к командным проектам контролируется путем назначения включения каждого участника команды в группу того или иного проекта. При этом можно добавлять членов команд отдельно или [добавлять группы, определённые в Active Directory или в рабочей группе](#).

Во время создания командного проекта для него создается четыре группы по умолчанию, независимо от выбранного шаблона процесса. Это группы Читатели, Участники, Администраторы сборки и Администраторы проекта. По умолчанию [каждая из этих групп имеет набор разрешений, определенных](#) для них. Эти разрешения определяют то, что могут делать участники данных групп.

5. Планирование работы

Практика

Начать планирование работы можно с доски задач и с определения спринтов. Эти функции доступны из Team Web Access.

Дополнительные сведения см. в разделе [Планирование и отслеживание проектов](#).

6. Добавьте исходный код.

Практика

Исходный код добавляется из [системы управления версиями Team Foundation](#) или [GIT](#).

7. Создание определений построения.

Практика

Определения сборок создаются из Visual Studio. См. раздел [Сборка приложения](#).

Создание командного проекта

И настройка его характеристик

+ Общее

+ Итерации (спринты)

+ Области

+ Безопасность

– Оповещение

+ Управление версиями

- Службы

+ Настройка перечня итераций (спринтов)

Большинство командных проектов имеют предопределенный набор путей итераций, основанных на шаблоне процесса. Можно переименовать этот набор или добавить в него пути. (<https://msdn.microsoft.com/ru-ru/library/ms181692.aspx>)

+ Настройка областей

Для группировки рабочих элементов по области продукта, функциональной области или области функций используйте пути к областям. Для группировки работы по спринтам, вехам или периодам времени, в течение которых она будет осуществляться, используйте пути итераций. Чтобы ограничить доступ к группе рабочих элементов, используйте путь к области или путь итерации (<https://msdn.microsoft.com/ru-ru/library/ms181692.aspx>)

Можно назначить разрешения на уровне пользователя или на уровне группы как для областей, так и для итераций. Разрешения ограничивают или разрешают доступ к рабочим элементам, тестовым случаям или планам тестирования. Также можно запретить или разрешить пользователям или группам управлять структурой проекта для области или итерации.

И прав доступа к этим областям

– *Безопасность*

... Мощная система, глубоко рассказывать некогда.

Пользователи, группы. В группы могут включены пользователи и группы.

Группы предопределённые и созданные.

Группы на уровне:

- экземпляра TFS
- коллекции проектов
- проекта

Права (разрешения): не установлены, наследуются, индивидуальные разрешения (да/нет).

– *Оповещение*

Задание. Мощная система, глубоко рассказывать некогда. Предлагается сделать обзор (реферат) с примерами из своей практики.


+ *Управление версиями (контролем версий)*

Папки

Разрешения на папки системы контроля версий

Настройка команды

The screenshot shows the Visual Studio interface for a project named 'Test2015_Scrum'. The browser address bar shows the URL: https://dmitriykalachev.visualstudio.com/DefaultCollection/Test2015_Scrum/_ad. The breadcrumb navigation is 'Control panel > DefaultCollection > Test2015_Scrum'. Below the breadcrumb is a tab bar with 'Overview' (selected), 'Iterations', 'Areas', 'Security', 'Alerts', and 'Version Control'. The 'Project profile' section on the left includes a placeholder icon for the team, the name 'Test2015_Scrum', and fields for 'Process' (set to 'Scrum') and 'Description' (with a placeholder 'Enter a description'). The 'Teams' section on the right shows a 'New team' button and a table with one team: 'Test2015_Scrum Team' with 1 member.

Team Name	Members
 Test2015_Scrum Team	1 member

Типовые группы коллекции проектов

Предварительно целесообразно в рамках коллекции проектов сформировать типовые группы на основе принятой системы распределения ролей

Control panel > DefaultCollection

Overview Security Build

Groups Users

Create VSO group

Search

- Project Collection Administrators
- Project Collection Build Administrators
- Project Collection Build Service Accounts
- Project Collection Proxy Service Accounts
- Project Collection Service Accounts
- Project Collection Test Service Accounts
- Project Collection Valid Users
- Аналитики
- Разработчики
- Тестировщики**

DefaultCollection > Тестировщики | Edit...

Permissions Members Member of

Administer build resource permissions	Not set
Administer Project Server integration	Not set
Administer shelved changes	Not set
Administer workspaces	Not set
Alter trace settings	Not set
Create a workspace	Inherited allow
Create new projects	Not set
Delete team project	Not set
Delete team project collection	Not set
Edit collection-level information	Not set
Make requests on behalf of others	Not set
Manage build resources	Not set
Manage process template	Not set
Manage test controllers	Not set
Trigger events	Not set
Use build resources	Not set
View build resources	Inherited allow
View collection-level information	Inherited allow
View system synchronization information	Not set

Типовые группы проекта

...


Добавление пользователя «со стороны»

А именно меня, что бы мог посмотреть, что же было сделано в рамках проекта.

Control panel > DefaultCollection > Test2015_Scrum > Test2015_Scrum Team

OverviewSecurityAlertsVersion ControlBuildA

Team Profile




Name
Test2015_Scrum Team








Description
The default project team.

Administrators
[+ Add](#)

Test2015_Scrum Team

Members Settings

Add... | 

Display Name	Use
 Аналитики	[De
 Разработчики	[De
 Тестировщики	[De
 Build Administrators	[Tes
 Project Administrators	[Tes
 Readers	[Tes
 DmitryKalachev	Dm

Members

Add...

ADD A USER

To add a user that is not currently known to Visual Studio Online, type their sign in address. If the member is known, just type their display name.

Identities [Browse](#) | [Check name](#)

No search results.

Save changes

Cancel

Check Name

Задание

...

02 | ОБЕСПЕЧЕНИЕ КАЧЕСТВА КОДА, КОНТРОЛЬ ВЕРСИЙ

В данном модуле будут рассмотрены: Основные концепции контроля версий, работа в режиме o, ветвления и слияния. Архитектурные инструменты Visual Studio, Архитектурные диаграммы и архитектурный обозреватель. UML Диаграммы, диаграммы зависимостей и слоев. Статический анализ кода и анализ кода.

А видео про это в уроке 03

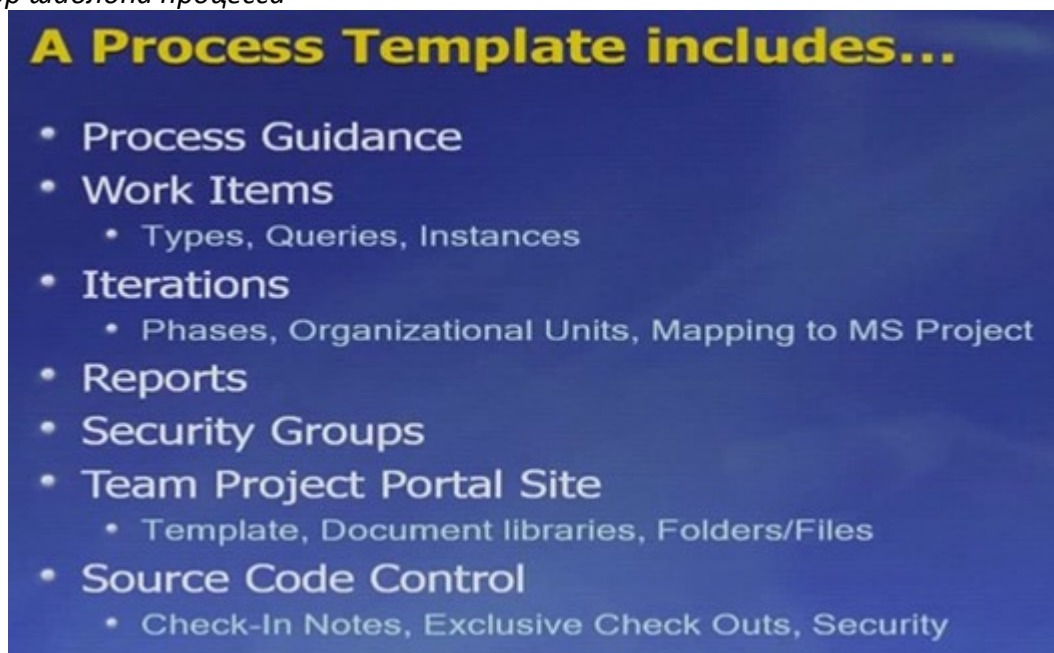
!!! обязательно посмотреть и начало урока 03

Теория

Создание проекта

!!! Так же см в пред разделе

Выбор шаблона процесса



О разных шаблонах см. «Работа с артефактами командного проекта, выбор шаблона процесса» (<https://msdn.microsoft.com/library/ms400752.aspx>)

...

Создание портала проекта (Share Point)

...рассматриваем только теоретически — какие возможности

Выбор системы контроля версий (начиная с 2013)

TFVC и Git ...

По желанию, но я могу прокомментировать только TFVC

Что получаем для управления

Запросы:

- Product Backlog - ...
- Итерация ..., Текущая итерация - ...
- ...

Добавьте исходный код

...с 7 минуты

См. <https://msdn.microsoft.com/ru-ru/library/ms181368.aspx>

При работе в команде или над сложными проектами практически во всех случаях следует работать с общей файловой системой с управлением версиями, чтобы упростить совместную работу и повысить прозрачность. Можно также воспользоваться преимуществами интегрированных [автоматизированных сборок](#), [планирования и отслеживания проектов](#) и [других возможностей управления жизненным циклом приложения](#), если работать с TFS ([локальным экземпляром](#) или [экземпляром в облаке](#)).

Подсистема контроля версий Team Foundation (TFVC) — это централизованная система управления версиями. Как правило, члены команды имеют на своих компьютерах разработки только одну версию каждого файла. Исторические данные ведутся только на сервере. Ветви основаны на путях и создаются на сервере.

TFVC позволяет команде вносить изменения двумя способами:

- [Серверные рабочие области](#): перед внесением изменений члены команды открыто извлекают файлы. Большинство операций требуют подключения разработчиков к серверу. (В прошлом команды не разрешали выполнять извлечение сразу нескольким пользователям, однако сейчас такой подход используется все реже.) Эта система облегчает блокировку рабочих процессов. Другие системы, работающие таким образом, включают Visual Source Safe, Perforce и CVS.
- [Локальные рабочие области](#): каждый член команды берет копию последней версии базы кода и работает с ней вне сети. Разработчики возвращают свои изменения и при необходимости разрешают конфликты. Ещё одна система, которая работает таким образом, — это Subversion.

Работа в Subversion мало отличается от работы в других централизованных системах управления версиями. Клиенты копируют [файлы](#) из хранилища, создавая локальные рабочие копии, затем вносят изменения в рабочие копии и фиксируют эти изменения в хранилище. Несколько клиентов могут одновременно обращаться к хранилищу. Для совместной работы над файлами в Subversion преимущественно используется модель *копирование — изменение — слияние*. Кроме того, для файлов, не допускающих слияние (различные бинарные форматы файлов), можно использовать модель *блокирование — изменение — разблокирование*.

[Дополнительные сведения о Team Foundation \(подсистеме контроля версий\)](#).

Что дает

Возможность

TFVC

Возможность	TFVC
Оповещения	Члены команды могут получать оповещения об извлечениях . Поскольку команда возвращает всю свою работу в централизованную систему, можно узнать, какой пользователь вернул тот или иной набор изменений и использовать сравнение , чтобы узнать, что именно изменено. Глядя на файл, можно добавить примечание , чтобы указать, кто изменил блок кода, а также когда это было сделано.
Контролируемость	Ветви на основе путей используются в основном в качестве долгосрочных конструкций, чтобы изолировать риск изменений между командами, работающими над различными компонентами и различными выпусками. Члены команды обычно создают дополнительную рабочую область для каждой ветви, над которой они работают.
Ветвление	Изменения в каждой ветви независимы друг от друга, поэтому возвращать их перед переключением между ветвями не нужно. Слияние между соседними ветвями требует слияния без базы. Можно получить визуализации структур ветвей и мест слияния наборов изменений. См. раздел Использование ветвей для изоляции рисков в системе управления версиями Team Foundation .
Сборки (автоматизированные с помощью TFSBuild)	Для сборки любого содержимого, необходимого в коллекции командных проектов, можно использовать все возможности TFSBuild.
Изменения	Члены команды могут одновременно изменять файлы на своих компьютерах разработки. Наборы изменений можно отправлять (возвращать) на сервер при их создании. Отправить изменения можно в любой момент. Однако процесс может быть прерван конфликтами . Можно изменить комментарий набора изменений после возврата набора изменений. Наборы изменений можно связывать с рабочими элементами и завершенными сборками.
??? Проверки кода в TFS	Да
??? Поддержка CodePlex	Да
Разрешение конфликтов	При получении, возврате, объединении или восстановлении из набора отложенных изменений может потребоваться разрешить конфликты . Все виды конфликтов можно разрешить в Visual Studio.
??? Хранение файлов	Можно возвращать большие двоичные файлы. Кроме того, можно использовать NuGet в качестве альтернативы или в сочетании с TFVC.
Файлы на клиентском компьютере разработки	Можно просматривать файлы с помощью обозревателя управления исходным кодом в Visual Studio, проводника Windows или командной строки .
Файлы на сервере	Каждый командный проект содержит все файлы в единственном корневом пути (например, \$/FabrikamTFVC). Можно применять разрешения на уровне файлов. Можно блокировать файлы . Можно просматривать файлы на веб-портале или с помощью обозревателя управления исходным кодом в Visual Studio. Ваш командный проект существует только на одном сервере.
Журнал	Журнал файлов не реплицируется на клиентском компьютере разработки, поэтому его можно просматривать только при наличии подключения к серверу. Можно просматривать журнал в Visual Studio и на веб-портале. Можно добавлять примечания к файлам, чтобы знать, кто изменил ту или иную строку и когда.
??? Управление работой на компьютере разработки	Страницы "Ожидающие изменения" и "Моя работа" .
??? Путь миграции	Git TF
Этапы контроля качества	Можно использовать сборки непрерывной интеграции , сборки с условным возвратом и политики возврата.

Возможность	TFVC
Откат изменений	Можно выполнить откат эффектов одного или нескольких наборов изменений
Масштаб	Можно работать над небольшими или очень крупными проектами с использованием локальных рабочих областей . Поддерживает очень большие проекты (миллионы файлов на ветвь и большие двоичные файлы) при использовании серверных рабочих областей .
Приостановка работы	Можно приостановить работу со страницы "Моя работа" или отложить изменения . Иногда необходимо отложить выполнение всей работы или её части. Система управления версиями способна избавить от части связанных с этим проблем и сократить время, которое теряется из-за таких перерывов. См. Приостановка работы
Пометка файлов тегами	Можно применить метки к версии одного или нескольких файлов из Visual Studio или из командной строки. У каждого файла могут быть метки, примененные к разным его версиям.
Инструменты	Клиентские инструменты: Visual Studio, Eclipse (с Team Explorer Everywhere) Серверные инструменты: TFS
Пользовательский интерфейс	Visual Studio: содержит все часто используемые функции и множество дополнительных функций. Веб-портал TFS: позволяет просматривать журнал базы кода и добавлять в него комментарии и заметки. Командная строка TF: устанавливается с Visual Studio. Используется для сложных, административных и других менее распространенных задач.
Совместимость Visual Studio	Можно использовать все поддерживаемые предыдущие версии Visual Studio .
Веб-портал	Можно выполнять поиск в базе кода (включая ветви), просматривать журнал, добавлять заметки и комментировать наборы изменений и наборы отложенных изменений, а также выполнять другие задачи, такие как загрузка избранных частей базы кода в виде ZIP-файла.

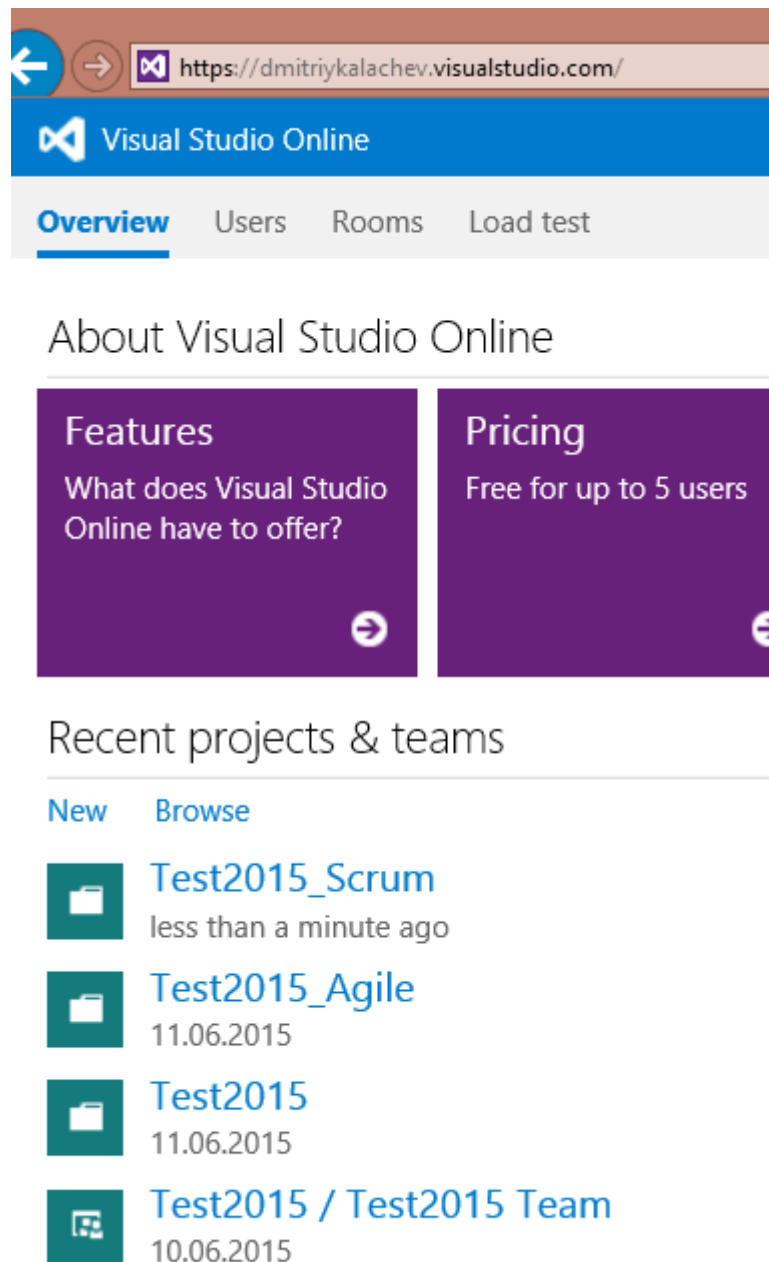
Практика

Виртуальные машины для обучения

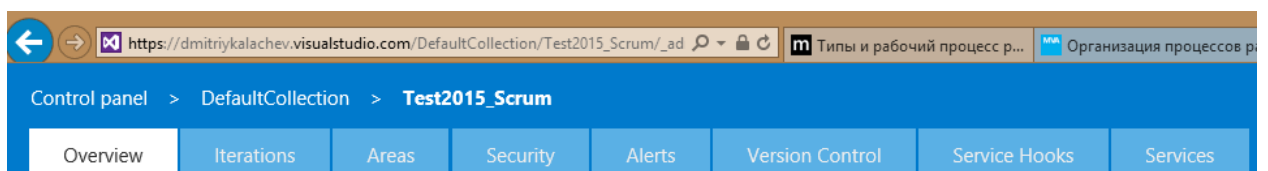
<http://blogs.msdn.com/b/briankel/archive/2013/08/02/visual-studio-2013-application-lifecycle-management-virtual-machine-and-hands-on-labs-demo-scripts.aspx>

Я буду пробовать все в online версии

Создание проекта



И настройка его характеристик



Project profile

- + Общее
- + Итерации (спринты)
- + Области
- + Безопасность
- Оповещение
- + Управление версиями
- Службы

Teams

+ Настройка перечня итераций (спринтов)

Большинство командных проектов имеют предопределенный набор путей итераций, основанных на шаблоне процесса. Можно переименовать этот набор или добавить в него пути. (<https://msdn.microsoft.com/ru-ru/library/ms181692.aspx>)

Задание длительностей итераций (и/или спринтов) – нормальная длительность спринта: две недели.

Control panel > DefaultCollection > Test2015_Scrum

Overview Iterations Areas Security

Iterations

Iterations

Select the iterations you want to use for iteration planning (sprint planning). Iterations will appear in your backlog view as iterations available for planning.

New New child

Iterations	Start Date
Test2015_Scrum	Set dates
<input checked="" type="checkbox"/> Sprint 1	
<input checked="" type="checkbox"/> Sprint 2	
<input checked="" type="checkbox"/> Sprint 3	
<input checked="" type="checkbox"/> Sprint 4	
<input checked="" type="checkbox"/> Sprint 5	
<input checked="" type="checkbox"/> Sprint 6	

+ Настройка областей

Области – некоторая иерархия маркеров для рабочих элементов.

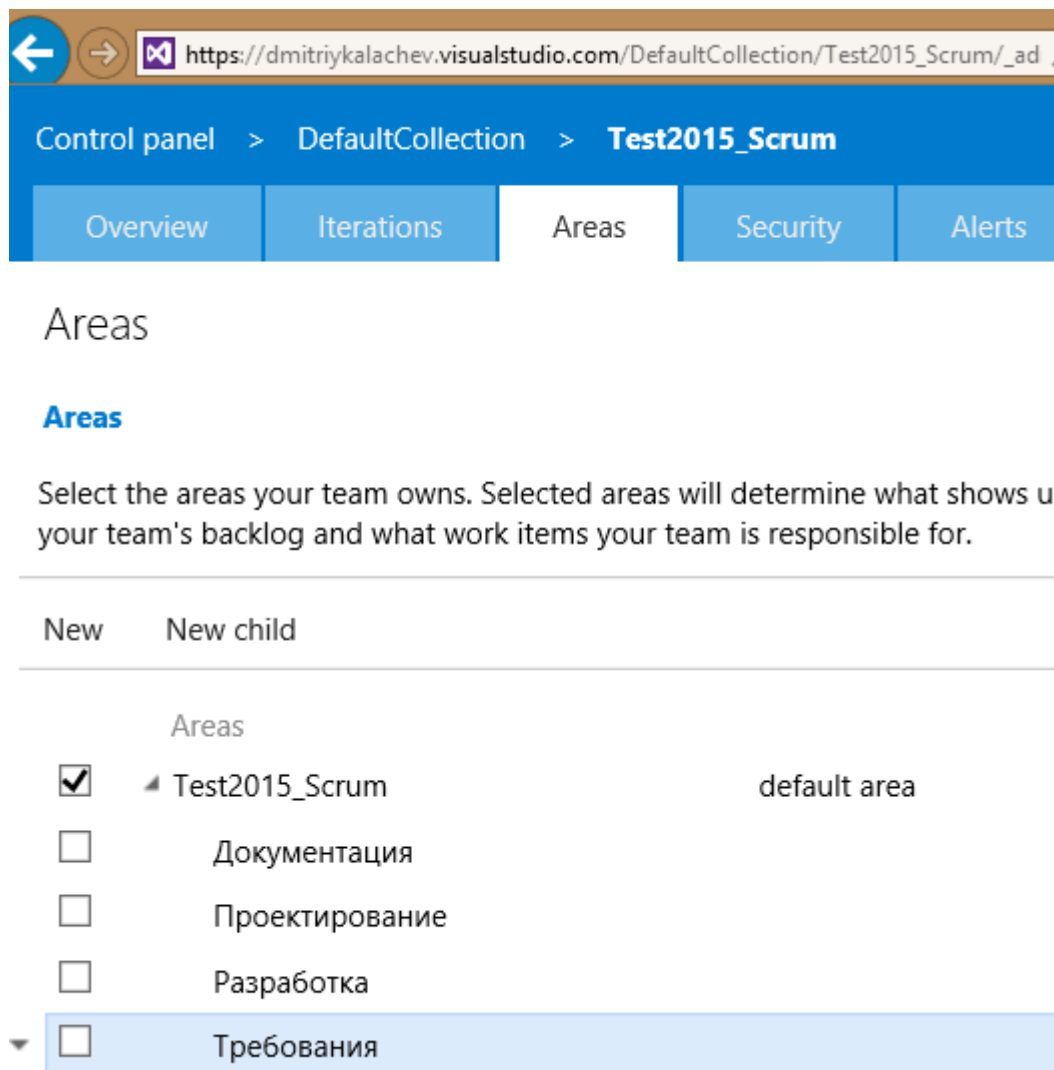
У WI только одна область.

Есть ещё теги для WI.

Для группировки рабочих элементов по области продукта, функциональной области или области функций используйте пути к областям. Для группировки работы по спринтам, вехам или периодам времени, в течение которых она будет осуществляться, используйте пути итераций. Чтобы ограничить доступ к группе рабочих элементов, используйте путь к области или путь итерации (<https://msdn.microsoft.com/ru-ru/library/ms181692.aspx>)

Можно назначить разрешения на уровне пользователя или на уровне группы как для областей, так и для итераций. Разрешения ограничивают или разрешают доступ к рабочим элементам, тестовым случаям или планам тестирования. Также можно запретить или

разрешить пользователям или группам управлять структурой проекта для области или итерации.



И прав доступа к этим областям

PERMISSIONS FOR ТРЕБОВАНИЯ

Add... ▾

VSO Groups

Аналитики
Project Collection Build Service Accounts
Project Collection Test Service Accounts
Build Administrators
Contributors
Readers
Users

ACCESS CONTROL SUMMARY

Shows information about the permissions being granted

Create child nodes	Allow
Delete this node	Allow
Edit this node	Allow
Edit work items in this node	Allow
Manage test plans	Inherited allow
Manage test suites	Inherited allow
View permissions for this node	Inherited allow
View work items in this node	Inherited allow

[Clear explicit permissions](#)

– Оповещение

Задание. Мощная система, глубоко рассказывать некогда. Предлагается сделать обзор (реферат) с примерами из своей практики.

Настройка команды

←

→

https://dmitriykalachev.visualstudio.com/DefaultCollection/Test2015_Scrum/_ad

Types and working

Types and working

Control panel > DefaultCollection > **Test2015_Scrum**

Overview

Iterations

Areas

Security

Alerts

Version Control

Project profile

Name
Test2015_Scrum

Process
Scrum

Description
Enter a description

Teams

New team

↺

Team Name

Members

Test2015_Scrum Team

1 member

Типовые группы коллекции проектов

Предварительно целесообразно в рамках коллекции проектов сформировать типовые группы на основе принятой системы распределения ролей

37

Control panel > DefaultCollection

Overview Security Build

Groups Users

Create VSO group

Search

- Project Collection Administrators
- Project Collection Build Administrators
- Project Collection Build Service Accounts
- Project Collection Proxy Service Accounts
- Project Collection Service Accounts
- Project Collection Test Service Accounts
- Project Collection Valid Users
- Аналитики
- Разработчики
- Тестировщики**

DefaultCollection > Тестировщики | Edit...

Permissions Members Member of

Administer build resource permissions	Not set
Administer Project Server integration	Not set
Administer shelved changes	Not set
Administer workspaces	Not set
Alter trace settings	Not set
Create a workspace	Inherited allow
Create new projects	Not set
Delete team project	Not set
Delete team project collection	Not set
Edit collection-level information	Not set
Make requests on behalf of others	Not set
Manage build resources	Not set
Manage process template	Not set
Manage test controllers	Not set
Trigger events	Not set
Use build resources	Not set
View build resources	Inherited allow
View collection-level information	Inherited allow
View system synchronization information	Not set

Планирование работы

См. Создание списка невыполненной работы <https://msdn.microsoft.com/ru-ru/library/ee518933.aspx>

??? Работа на доске канбан <https://msdn.microsoft.com/ru-ru/library/jj838789.aspx>

<https://www.visualstudio.com/get-started/work-from-the-kanban-board-vs>

??? Работа в спринтах (доска задач спринта) <https://msdn.microsoft.com/ru-ru/library/ee191595.aspx>

Формирование портфеля (backlog)

Создание WI ...

Тип – см. Ранее о модели ЖЦ

Название – ...

Состояние – ...

Причина – ...

Назначено на – ...

[Трудозатраты](#) Efforts (Scrum) – ...

[Ценность для бизнеса](#) BusinessValue (Scrum) – ...

Приоритет Priority – ...

Назначение на итерацию

... Статус: Committed Зафиксировано ...

Добавьте исходный код

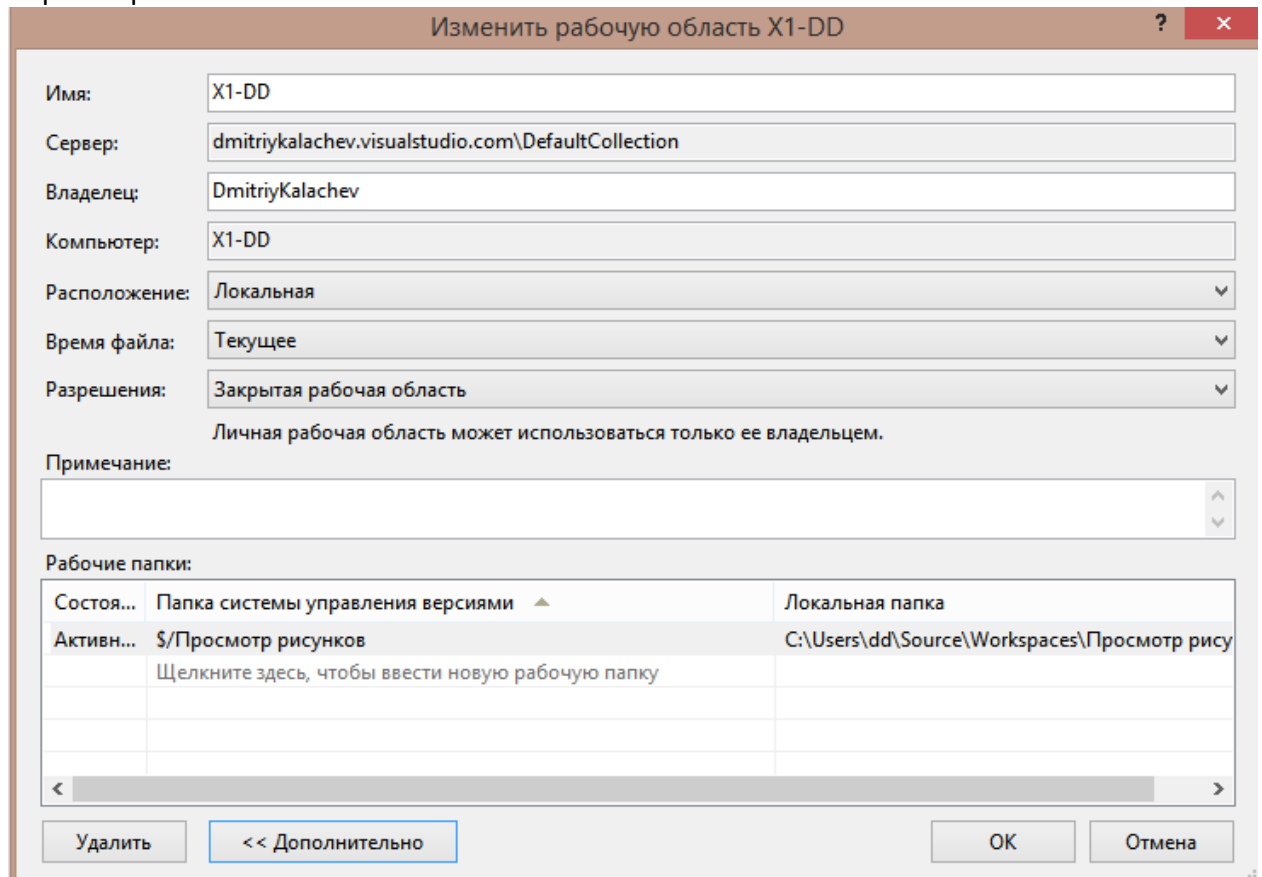
Систему контроля версий для проекта мы выбрали при создании командного проекта.

Примеры кода [https://msdn.microsoft.com/ru-ru/library/dd492171\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/dd492171(v=vs.110).aspx)

Настройка компьютера разработки и начало работы

Потратьте несколько минут, чтобы настроить компьютер разработчика для получения всех преимуществ от использования базы исходного кода, помещённой в систему управления версиями.

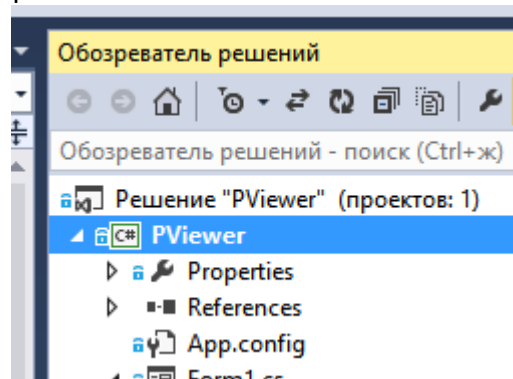
- Выбор СУВ – TFS online ...
- Выбор проекта в TFS
- Настройка рабочей области



Разработка приложения в базе кода с управлением версиями

В большинстве случаев вам не придётся задумываться о работе системы управления версиями. Эта система окажет вам поддержку в случаях, когда необходимо анализировать внесённые изменения и управлять ими.

- ...Создать первую версию приложения...



- Добавить решение в СУВ.
управления версиями+Добавить в СУВ

контекстное меню Система

- Сохранить изменения – Check-In (Возврат)
- ...
- ...

Задание

...

03 | ТЕСТИРОВАНИЕ ПО И СБОРКА

К уроку. В данной части курса мы рассмотрим модульное тестирование, поддержка различных сред модульного тестирования. Конфигурирование автоматических сборок. Обзор Microsoft Test Manager.

К фильму. Модульное тестирование Поддержка различных сред модульного тестирования. Конфигурирование автоматических сборок. Обзор Microsoft Test Manager.

0 мин – СУБ, связь изменений с рабочими элементами, Статический анализ, Ветвление и слияние
 17 мин – Архитектура решений. Есть только для Ultimate – создание диаграмм моделирования в т.ч. и UML
 30 мин – вернулся к доске
 34 – модульные тесты, самый хвостик в **следующем видеоуроке**
 ??? а где про сборки в кино ?

Обязательно посмотреть начало следующего видеоурока

Теория

Статический анализ кода

...че тут говорить – см. урок

Система управления версиями

Это по существу материал урока 02

Участие в работе команды

Возвращайте в систему файлы с внесёнными изменениями так, чтобы ваша команда могла использовать созданную вами функциональность в процессе построения, тестирования и создания релизов.

Политики возврата кода на сервер

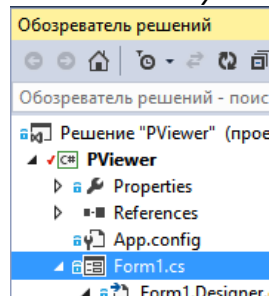
Обязательность комментария к изменениям

Обязательность связывания с рабочими элементами

Для трассируемости

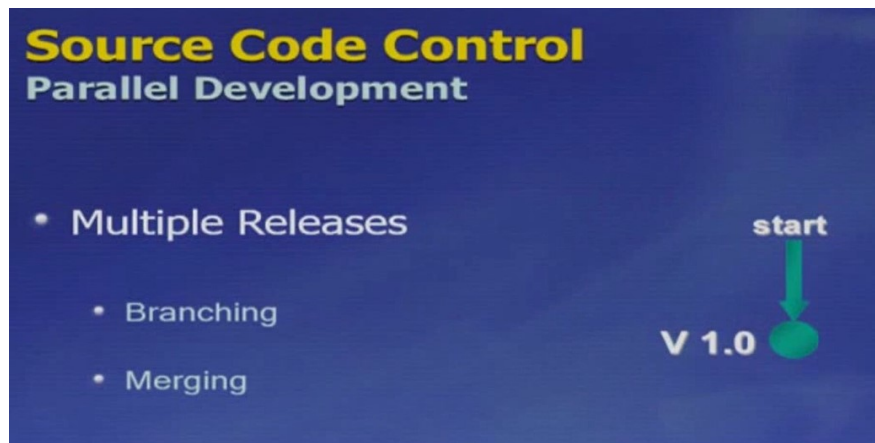
Для автоматического закрытия рабочего элемента при возврате кода на сервер

Возможность в студии увидеть историю изменений

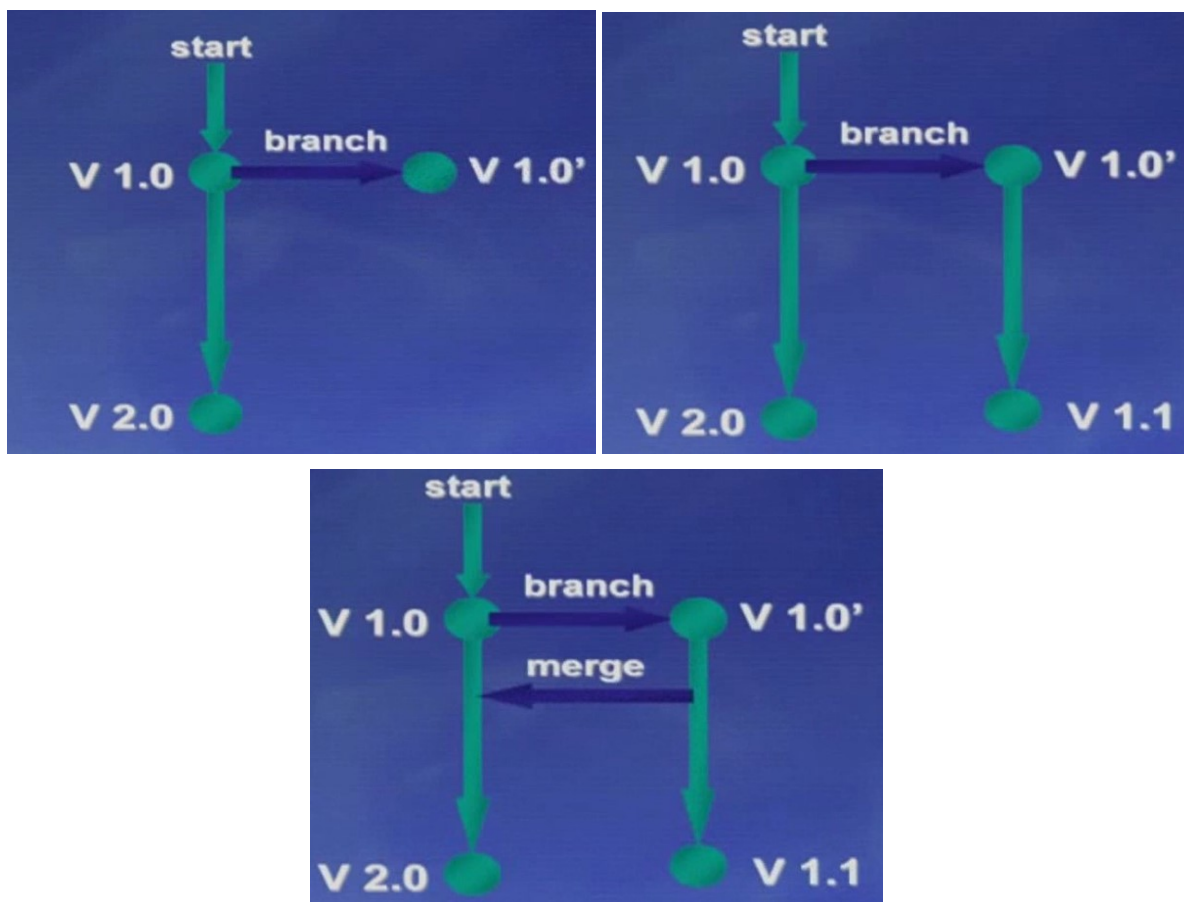


контекстное меню Система упр версиями + Заметки (Annote)

Выбор объёма анализа кода



Обязательно маркирование исходников версии 1.0



Модульное тестирование

с 34 минуты — модульные тесты

Автоматические сборки

По плану д.б в уроке 03 и там же тестирование прохождения темы. Но показано в кино 04, начиная с 3:30 по 12:30 мин

Определение сборки стало проще. См. раздел [Использование шаблона по умолчанию для процесса сборки](#)

Полезно иметь отдельную машину (рекомендуется виртуальную) для использования в качестве build-server. См. [Развертывание системы сборки](#). Чтобы использовать Team Foundation Build, потребуется один или несколько серверов сборки.

Если коллекция командных проектов размещена в [Visual Studio Online](#), вместо развертывания собственной системы сборки можно использовать [размещенный контроллер сборки](#).

[Использование контроллера сборки с коллекцией командных проектов, размещенной в Visual Studio Online](#)

Если коллекция командных проектов размещена в [Visual Studio Online](#), можно не осуществлять развертывание собственного контроллера сборки и использовать вместо него размещенный контроллер сборки (Hosted Build Controller). Чтобы использовать размещенный контроллер сборки, выделите его при [определении процесса сборки](#). См. раздел [Использование размещенного контроллера сборки](#).

Использование локального контроллера сборки. Если в процессе сборки требуются ресурсы за пределами [возможностей размещенного контроллера сборки](#), к размещенной коллекции командных проектов можно подключить локальный контроллер сборки.

Практика

Статический анализ кода

Запустить ...

Главное меню Анализ + ...

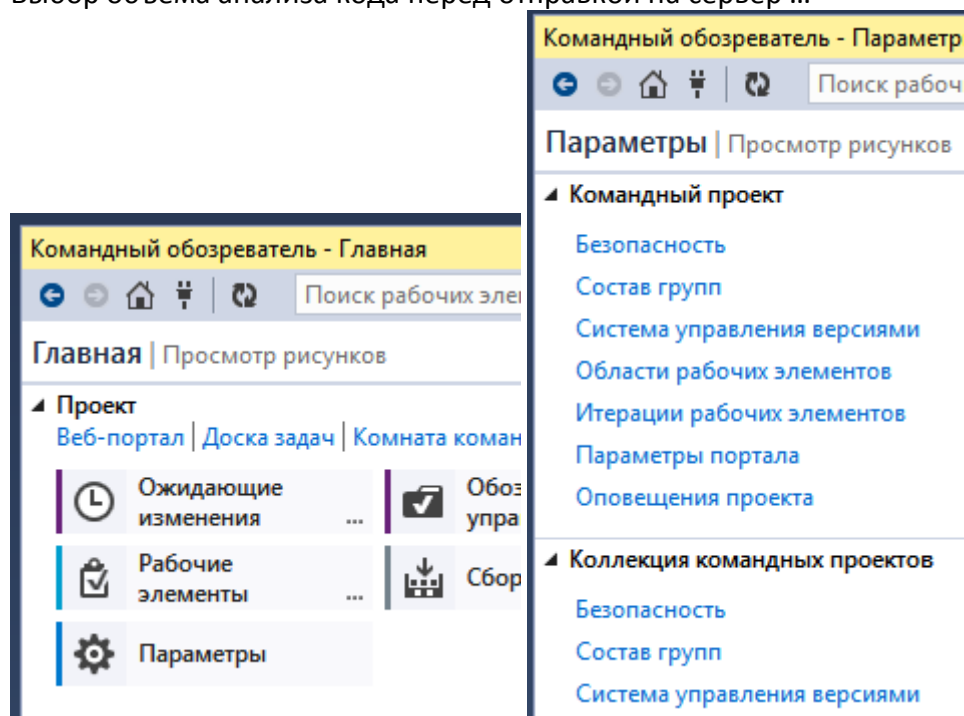
Настроить объем проверок ...

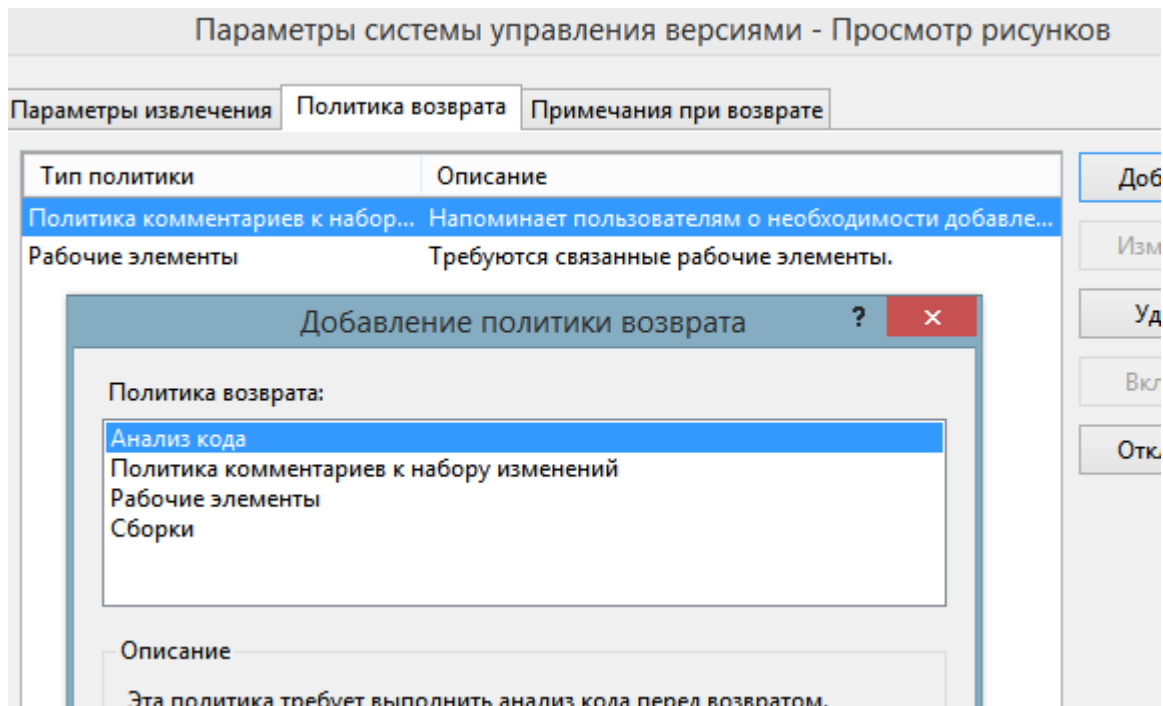
Главное меню Анализ + ...

Система управления версиями

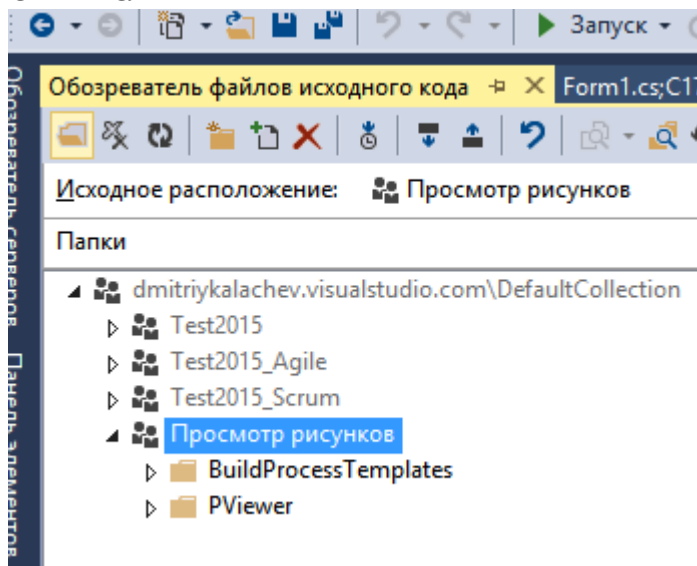
Политики возврата кода на сервер

Выбор объема анализа кода перед отправкой на сервер ...

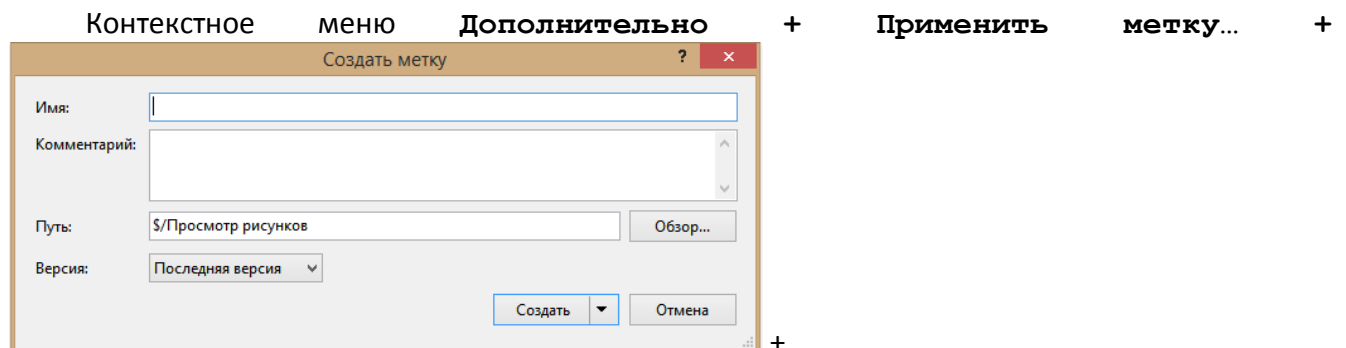




Ветвления и Слияния



Маркирование



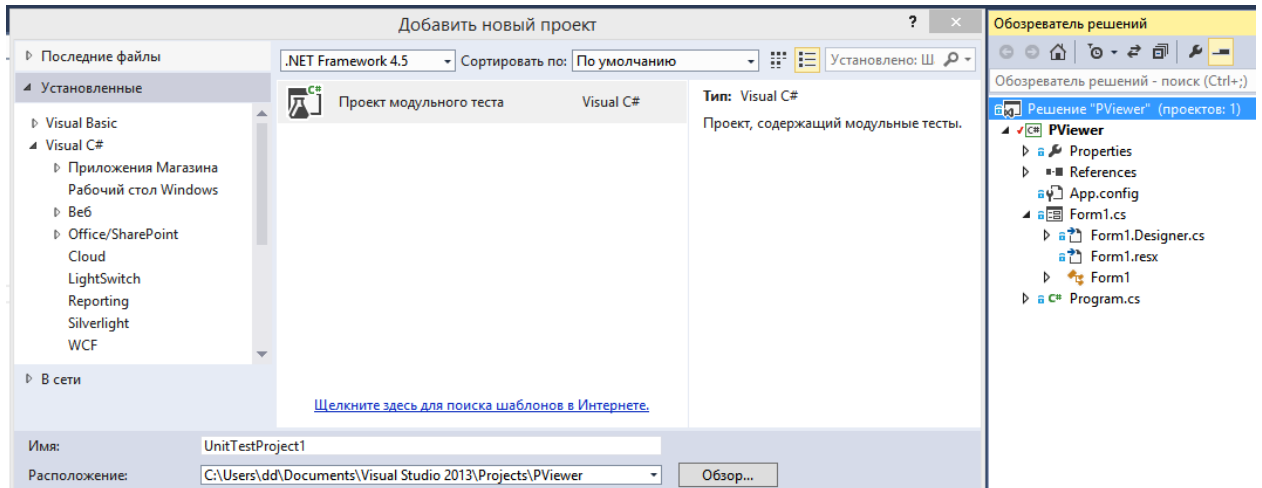
[https://msdn.microsoft.com/ru-ru/library/vstudio/ms181439\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/vstudio/ms181439(v=vs.110).aspx)

Ветвление

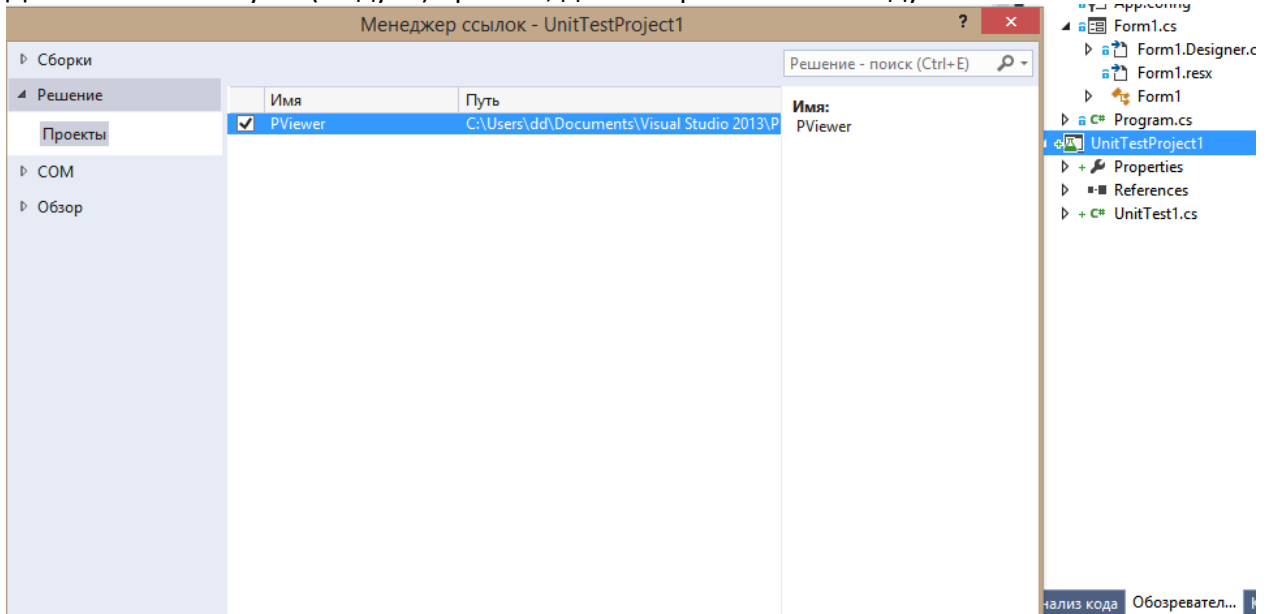
Контекстное меню **Ветвление и объединение** + **Ветвь...**

Модульное тестирование

Создаём проект модульных тестов для нашего проекта. В окне **Обозреватель решений**.



Добавляем ссылку на (модуль) проекта, для которого пишем модульные тесты.

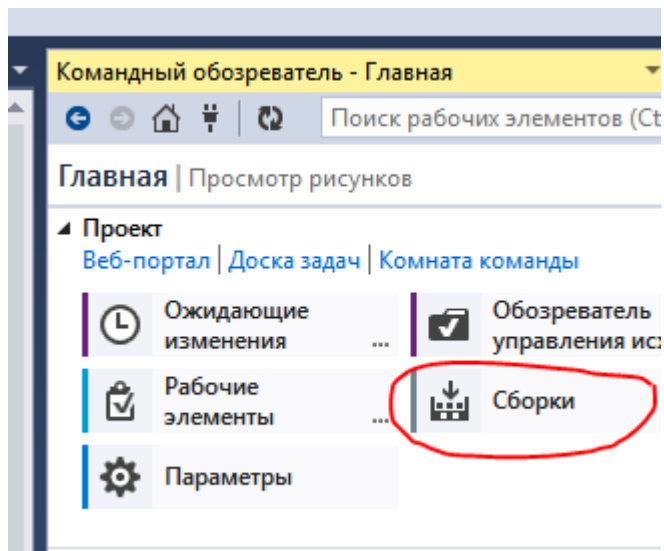


???...не придумал мод тестов для просмотривальщика картинок

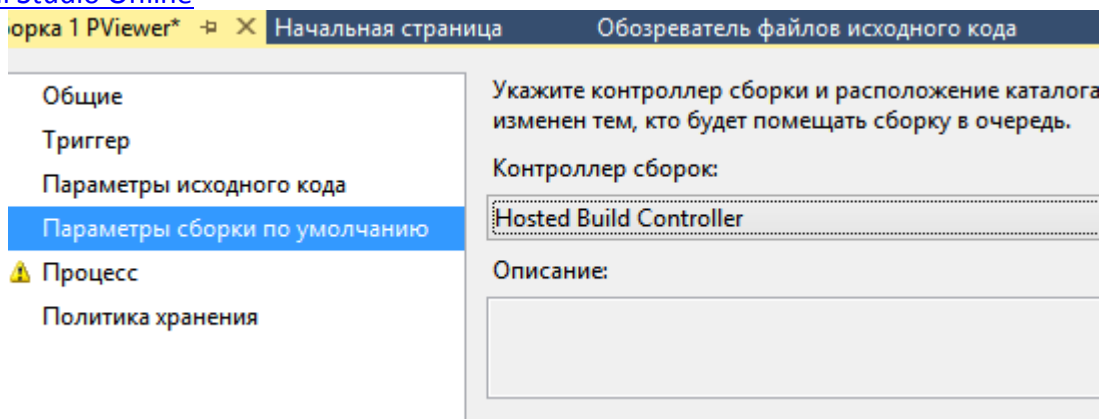
Автоматические сборки

Создание и настройка

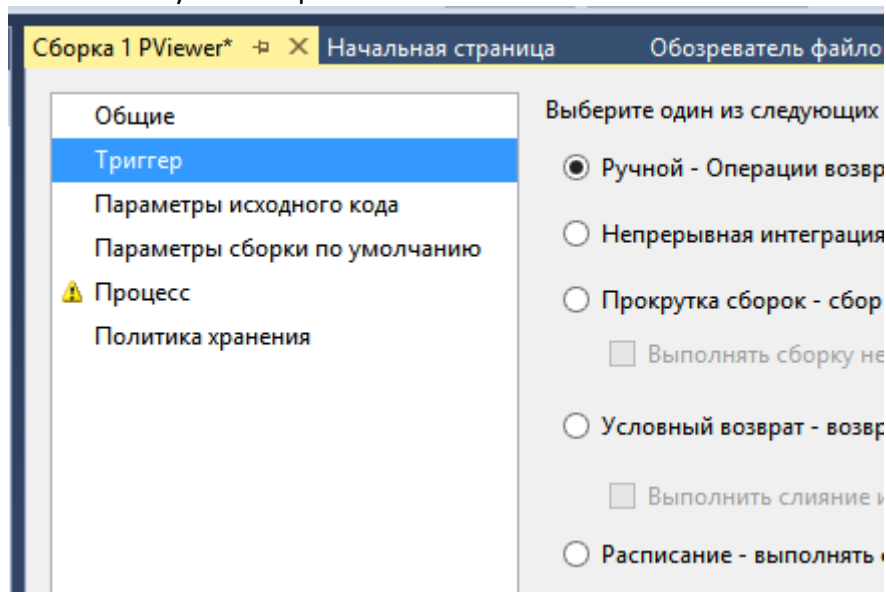
В командном обозревателе VS пункт Сборки и Создать определение сборки, дать имя



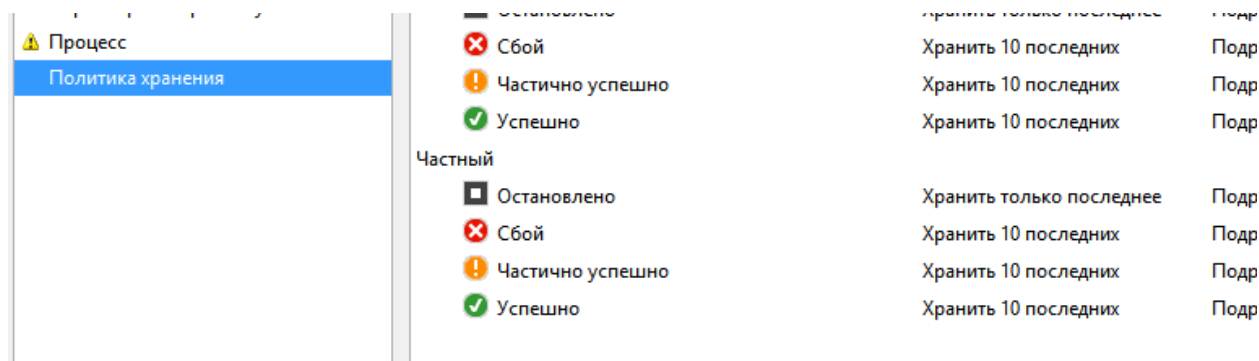
[Использование контроллера сборки с коллекцией командных проектов, размещенной в Visual Studio Online](#)



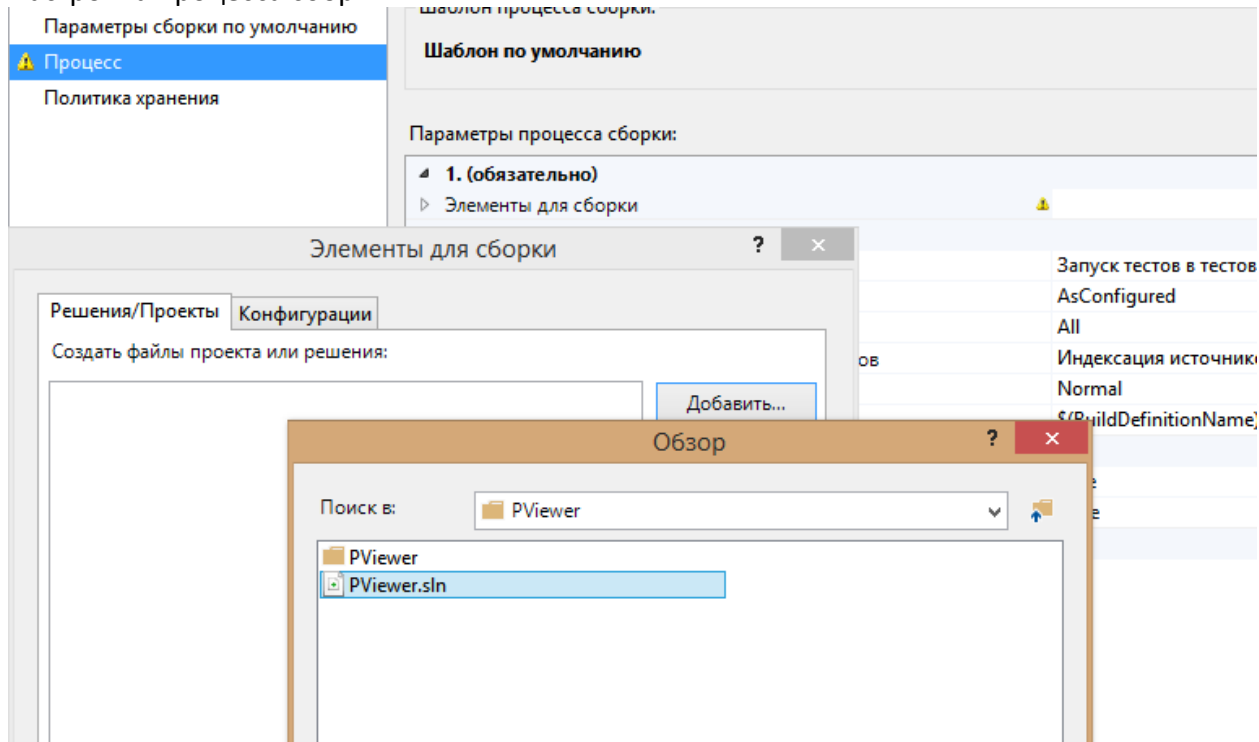
Условие запуска построения



Есть такая штука как **Политика хранения**, что бы не хранить все построения

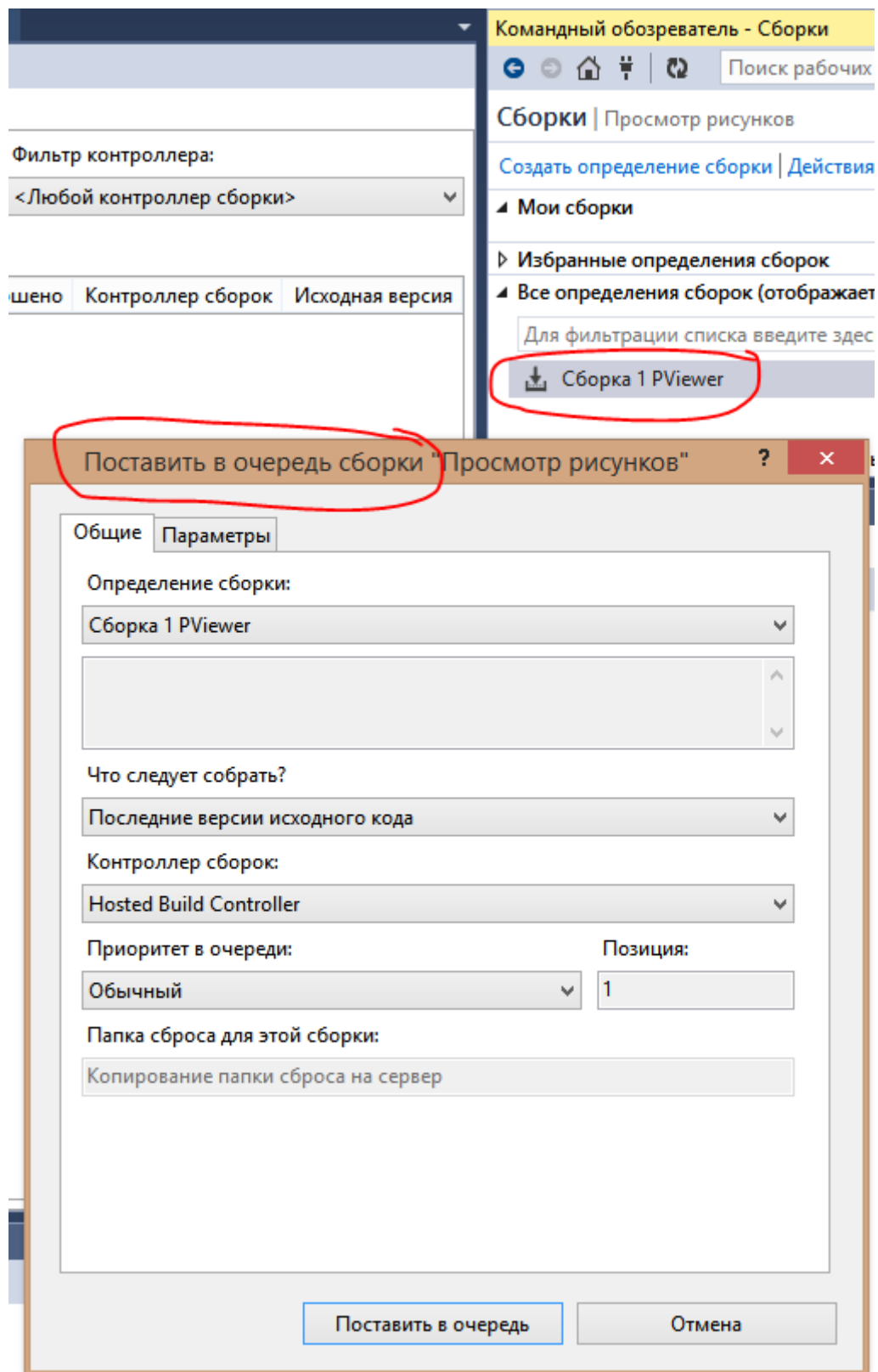


Настройка процесса сборки



Запуск сборки

В контекстном меню пункт **Поставить в очередь...**



- Увеличивает номер сборки
- Стягивает исходники
- Компилирует и компоует
- Запускает модульные тесты

По окончании во вкладке Завершённые, в контекстно меню Открыть, то получим:

Обозреватель сбо...осмотр рисунков Сборка 1 PViewer Начальная страница Обозреватель файлов исходного кода Сборка Сборка 1 PViewer_20150709.1

✓ Сборка 1 PViewer_20150709.1 - Сборка успешно завершена

Просмотреть сводку | Просмотреть журнал | Открыть папку для размещения | Диагностика | <Качество не назначено> | Действия

DmitriyKalachev инициировал(а) Сборка 1 PViewer (Просмотр рисунков) для Набор изменений 17
Выполнялось 58 с (Hosted Build Controller), завершено 4,5 мин назад

Последнее действие

Сборка последний раз была изменена Elastic Build (DmitriyKalachev) 4,5 мин назад.

Сводка запроса

Запрос 1, автор запроса: DmitriyKalachev, запрос получен 6,3 мин назад, Завершено

Сводка

Debug | Any CPU

Ошибок: 0, предупреждений: 0

► скомпилировано: \$/Просмотр рисунков/PViewer/PViewer.sln

Нет результатов теста

Нет результатов по объемам протестированного кода

Затронутые тесты

Тесты не были затронуты

Задание

...

04 | АВТОМАТИЗИРОВАННОЕ ТЕСТИРОВАНИЕ

К уроку. В данном модуле мы рассмотрим конфигурирование сред тестирования, автоматические тесты. Тестовые планы. Исследовательское тестирование, устранение барьеров взаимодействия тестировщика и разработчика.

К фильму. Конфигурирование сред тестирования. Автоматические тесты. Тестовые планы. Исследовательское тестирование. Устранение барьеров взаимодействия тестировщика и разработчика.

Первая минута - тупо продолжение предыдущего фильма про модульные тесты.

Кино 04, начиная с 3:30 по 12:30 мин о сборках, так что смотреть про авт тестирование начиная с 12:30

Теория

Документация тестирования: тест-план, тестовые спецификации

Основа для тестовых спецификаций:

- Требования – черный ящик
- Задачи – белый ящик

Минимум ручное тестирование

Автоматизация важно для регрессионного тестирования

Тестирование приложения с помощью Microsoft Test Manager

!!! Test Manager Test Manager устанавливается с Visual Studio Test Professional и Visual Studio Enterprise.

<https://msdn.microsoft.com/ru-ru/library/jj635157.aspx>

Практика

Планирование ручных тестов с помощью Team Web Access

<https://msdn.microsoft.com/ru-ru/library/dd380763.aspx>

Задание

Задание. Сделать несколько ручных тестов

...

05 | ДИАГНОСТИКА, ЭКСПЛУАТАЦИЯ, ОТЧЁТНОСТЬ

К уроку. В данном модуле мы рассмотрим диагностику и отладку с помощью отладочных символов. Настройку шаблонов процессов и работа с отчётностью.

До 25 минуты ещё предыдущая тема

Теория

...

Практика

С 26 минуты настройка построений

С 31 минуты – о нагрузочных тестах !!! только для Visual Studio Ultimate

С 58 минут несколько слов о TFS Power Tools

С 1:00 час о том что такое шаблон процесса

1:02 «отчётность и тому подобные вещи» - фигня

Задание

...