

Алексей Фёдоров

Знакомство с

Microsoft<sup>®</sup>

Visual Studio<sup>®</sup>

Team System 2008



Microsoft<sup>®</sup>

Visual Studio<sup>®</sup> Team System 2008

**Алексей Федоров**

# **Знакомство с**

Microsoft®

**Visual Studio®  
Team System 2008**

Алексей Федоров

**Знакомство с Microsoft®  
Visual Studio® Team System 2008**

Алексей Федоров — сотрудник отдела стратегических технологий  
ООО «Майкрософт Рус» (alexEIF@microsoft.com)

Подготовлено к печати издательством «Русская Редакция»  
123290, Москва, Шелепихинская наб., д. 32  
тел.: (495) 256-6691, тел./факс: (495) 256-7145  
e-mail: info@rusedit.com, <http://www.rusedit.com>

 РУССКАЯ РЕДАКЦИЯ

© Федоров А. А., 2007  
© «Русская Редакция», 2007

# Содержание

Роли в программном проекте .....	5
Командная разработка .....	7
Visual Studio Team System Web Access .....	9
Microsoft Visual Studio 2005 Team Foundation Server Power Tool .....	11
«Ролевые» версии Visual Studio .....	12
Architecture Edition — Visual Studio для архитекторов .....	12
Development Edition — Visual Studio для разработчиков .....	14
Database Edition — Visual Studio для разработчиков и администраторов баз данных .....	16
Ролевой подход .....	17
Test Edition — Visual Studio для тестировщиков .....	18
Запись и воспроизведение тестов .....	18
Нагрузочное тестирование .....	19
Управление тестированием .....	19
Visual Studio Team System в развитии — проект «Rosario» .....	20
Механизмы расширения Visual Studio .....	23
Макросы .....	23
Дополнительные модули .....	23
Механизмы VS Package .....	24
Расширения Visual Studio Team System .....	24
Visual Studio 2008 Shell .....	24
Visual Studio Team System. Полезные ссылки .....	25
Общие ресурсы .....	25
Team Foundation Server .....	25
Для архитекторов .....	26
Для разработчиков .....	26
Для разработчиков и администраторов баз данных .....	26
Для тестировщиков .....	26
Visual Studio 2008 .....	27
Visual Studio «Rosario» .....	27
Механизмы расширения Visual Studio .....	27
Партнерские решения .....	27
Приложение. Аппаратные и программные требования .....	28
Рекомендованные аппаратные конфигурации .....	31

*С появлением в линейке средств разработки продукта под названием Visual Studio 2005 Team System компания Microsoft вышла за рамки базового процесса разработки — редактирование-компилирование-отладка и предложила набор инструментов, ранее доступных только от сторонних производителей. Средства, включенные в состав Microsoft Visual Studio 2005 Team System, предназначены для более ранних шагов цикла разработки, таких как дизайн и архитектура, а также шагов, выполняемых после непосредственной разработки — тестирование и развертывание. Включение подобных средств в состав Visual Studio 2005 позволяет использовать его на всех этапах создания программного продукта.*

*Новая версия — Visual Studio Team System 2008 продолжает развитие семейства продуктов для управления всем жизненным циклом создания приложений и содержит новые и улучшенные средства, процессы и руководства, которое помогут улучшить совместную командную работу и сделать ее более эффективной. Инструменты, входящие в состав Visual Studio Team System 2008, позволяют наладить более эффективные коммуникации между членами проектной группы и заказчиками, наладить эффективную совместную работу, обеспечить ожидаемое заказчиками качество кода, используя расширенные средства контроля качества, получить представление об активностях в рамках проекта и приоритетах, которые позволят принимать решения, основываясь на данных, предоставляемых в реальном времени.*

## Использование Visual Studio Team System в Microsoft

Проектные команды, отвечающие за разработку ключевых продуктов Microsoft (Windows, Office, SQL Server, Development Tools), активно используют Visual Studio Team System. Помимо этого, подразделение Microsoft IT также использует этот продукт для создания проектов для внутреннего использования. Вот лишь несколько цифр. В Microsoft установлены более 20 копий Team Foundation Server (большинство — это Beta 2 Team Foundation Server 2008, остальные — ранние версии Visual Studio Team System «Rosario»), на которых поддерживается более 730 проектов и более 5600 активных пользователей.

## Роли в программном проекте

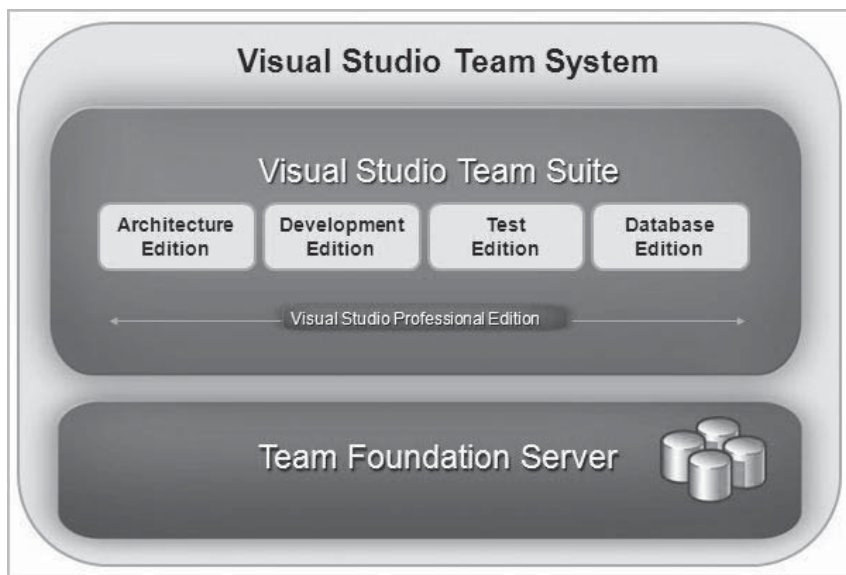
Задача любой формализованной методологии — структурировать процесс разработки программного обеспечения. В частности, любая методология определяет роли в программном проекте. Наиболее часто используются следующие роли:

- **Архитектор** отвечает за высокоуровневую структуру приложения, требования, ключевые технологии для реализации, взаимодействие приложения с другими приложениями, инфраструктурой и т. п.
- **Разработчик** занимается написанием и отладкой кода. Предыдущие версии Visual Studio, а также продукты, входящие в состав Visual Studio 2008 — Visual Studio Standard Edition и Visual Studio 2008 Professional Edition предназначаются, в первую очередь, для этой роли.
- **Тестировщик** отвечает за обнаружение ошибок, их исправление; должен убедиться в том, что приложение соответствует изначальным требованиям (функциональным и нефункциональным).
- **Разработчик баз данных** занимается разработкой приложений, использующих базы данных (работа со схемой базы данных, создание хранимых процедур и т. п.).
- **Менеджер проекта** отвечает за ведение проекта, проектный офис, метрики, сроки и т. п.

Важно отметить, что каждая роль не обязательно соответствует наличию физического члена команды. В небольших проектах возможно совме-

щение ролей. Тем не менее, даже если один человек выполняет несколько ролей, для выполнения каждой задачи он использует разные средства.

Visual Studio Team System 2008 — это продукт, который призван облегчить задачи, выполняемые каждой из перечисленных выше ролей. В состав продукта входят четыре т. н. ролевых издания, построенных поверх Visual Studio 2008 Professional Edition — **Architecture Edition**, **Development Edition**, **Database Edition** (это издание впервые появилось в Visual Studio Team System 2008) и **Test Edition** — вместе эти издания называются Team Suite, а также серверный продукт, обеспечивающий совместную работу над проектами (совместный доступ к ресурсам, контроль версий, порталы на базе SharePoint, отчеты, слежение за элементами проекта и т. п.) — **Team Foundation Server**.



Visual Studio Team System 2008

## Командная разработка

В основе Visual Studio 2005 Team System находится серверное приложение под названием Team Foundation Server, состоящее из базы данных на SQL Server (используемой для хранения проектных данных) и компонентов среднего уровня на базе Windows Server 2003, ASP.NET и Windows SharePoint Services. Основные компоненты Team Foundation Server показаны на следующей диаграмме.

Задача Team Foundation Server — обеспечить инфраструктуру для совместной работы и обмена информацией между группами архитекторов,

разработчиков, тестировщиков и менеджеров проекта. Клиентским компонентом является Team Foundation Client, который входит в состав всех продуктов семейства Team Edition, а также может использоваться продуктами Microsoft (Excel, Project) или продуктами других производителей.



#### Компоненты Team Foundation Server

Как видно из приведенной диаграммы, компоненты Team Foundation Server отвечают за управление требованиями, слежение за элементами проекта, обеспечивают автоматизацию сборки проекта, генерацию отчетов, а также интеграцию с другими подсистемами продуктов семейства Team System.

В состав Team Foundation входят средства создания проектов (Project Creation Wizard) с возможностью выбора шаблонов (по умолчанию используется шаблон MSF For Agile Development), задания типов артефактов, итераций, отчетов, контроля версий, адреса проектного портала и т. п., средства контроля версий, пришедшие на смену Visual SourceSafe и рассчитанные на объемные проекты с большим числом разработчиков. В процессе работы над проектом собирается большой объем информации (для этих целей служит SQL Server 2005).

Эти данные могут быть отображены с помощью более чем 50 типов отчетов (на базе SQL Reporting Services), входящих в состав продукта (например, Work Item History — Record-Count, TransitionCount, RemainingWork, CompletedWork, BaselineWork, Code Churn — TotalLines, LinesAdded, Lines-Modified, LinesDeleted, Test Results — TotalTests, TestsFailed, AverageDuration и т. д.) и опубликованы на портале проекта (Project Portal на основе Windows SharePoint Services или Microsoft Office SharePoint Server 2007) через соответствующие веб-компоненты.

Проектная деятельность может вестись согласно определенным методологиям. В состав продукта входит поддержка двух методологий Microsoft — MSF For Agile Development и MSF For CMMI Process Improvement. Также поддерживается «классическая» методология Microsoft Solutions Framework. Помимо этого можно использовать методологии сторонних поставщиков или создавать собственные, отражающие подходы к созданию продуктов, сформированные внутри конкретной компании.





В Visual Studio 2008 Team Foundation пополнился рядом новых возможностей, а также получил ряд улучшений в существующих компонентах.

**Team Foundation Build** — средство для управляемой сборки проектов. Теперь описания процесса сборки (Build Definitions) заменяют типы сборки (Build Types) из Microsoft Visual Studio 2005 Team System. В отличие от типов сборки, описания процесса сборки могут быть изменены в Team Explorer. Помимо этого из пользовательского интерфейса можно остановить выполняющуюся сборку и полностью удалить завершённый процесс. Также при создании нового описания процесса сборки или редактировании существующего можно установить триггер для процесса сборки. Можно использовать сборку по требованию, многократную сборку и интеграцию, при которой каждое освобождение (check-in) кода приводит к запуску процесса сборки. Также можно задать время ожидания между выполнениями сборок при многократных сборках. В Visual Studio 2008 Team Foundation Server появился ряд новых свойств для настройки процесса сборки, включая возможность настройки сборок для кода на C++. В состав Team Foundation Build входит ряд новых шаблонов, которые можно использовать для создания собственных процессов сборки.

**Team Foundation Source Control** — это входящее в состав Team Foundation средство контроля версий. В Visual Studio 2008 появились следующие возможности:

- Полное удаление файлов (команда Destroy), находящихся под управлением системы контроля версий.
- Автоматическое получение самой последней версии файла при выполнении операции check-out.
- Возможность аннотирования исходных файлов — построчный просмотр внесённых изменений, а также даты внесения этих изменений.
- Возможность сравнения двух папок на сервере, двух локальных папок или папки на сервере с папкой на локальном компьютере с использованием контроля версий. Пользователь получает информацию о различиях — отсутствующих файлах, файлах, в которые были внесены добавления, удаления, а также о конфликтах между файлами в разных папках.

**Team Foundation Work Item Tracking** — это механизм слежения за элементами проекта. В Visual Studio 2008 появились существенные улучшения при отслеживании элементов проекта под большими нагрузками. По сравнению с Team Foundation Server 2005, пропускная способность уве-

личилась вдвое, существенно снизилось время, требуемое для завершения отдельных операций и снизилось потребление процессорных ресурсов сервера, на котором располагается слой базы данных Team Foundation Server. Масштабируемость Team Foundation Server 2008 также была увеличена — так что время отклика для большинства операций слежения существенно сократилось, даже при высоких нагрузках на сервер. Это особенно заметно при работе проектных команд, состоящих из более чем 500 чел. Таким образом крупные компании могут использовать больше проектных элементов и следить за ними на существующих серверных конфигурациях, работающих под управлением Team Foundation Server 2008.

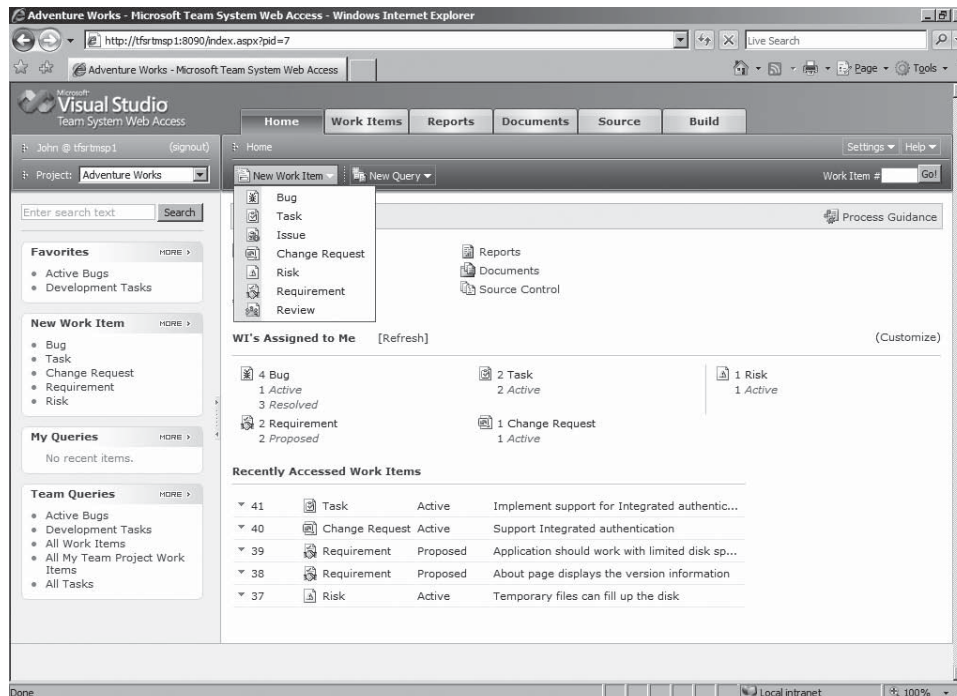
**Управление Team Foundation Server** — в новой версии сервера добавление большого числа пользователей не приводит к задержкам в работе и другим связанным с этим проблемам. Тогда как число поддерживаемых пользователей не изменилось, скорость синхронизации пользователей между Active Directory и Team Foundation Server существенно повысилась.

Как мы отметили выше, Team Foundation служит в качестве основы для работы различных участников проекта — менеджеров, архитекторов, разработчиков и тестировщиков, организации обмена информацией между ними. Для каждой конкретной роли — архитектора, разработчика, разработчика и администратора баз данных, тестировщика предлагается специализированная, ролевая версия Team System, которую мы и рассмотрим ниже.

## Visual Studio Team System Web Access

Team System Web Access (продукт, ранее известный под названием TeamPlain Web Access и приобретенный у компании devBiz Business Solutions) представляет собой веб-интерфейс к Visual Studio 2005 Team Foundation Server. Team System Web Access доступен для бесплатной загрузки (<http://go.microsoft.com/?linkid=7148426>) всем пользователям Team Foundation Server и будет включен в Visual Studio 2008 Team System.

К ключевым характеристикам продукта относятся возможность добавления новых элементов программного проекта и редактирование существующих, использование любых типов элементов проекта, включая нестандартные, задаваемые пользователями, возможность добавления новых элементов в запросы и редактирование существующих запросов, наличие средств просмотра, загрузки, выгрузки, выполнения операций check-in и check-out над документами, расположенными на проектном портале на SharePoint Server, просмотр отчетов, экспорт отчетов в PDF или Excel, просмотр репозитариев системы контроля версий, загрузка файлов, просмотр наборов изменений (changesets), различий между файлами, истории изменений и представлений с аннотациями, возможность просмотра результатов сборки проекта, запуск и остановка процесса сборки, а также поиск по ключевым словам в элементах проекта.



### Team System Web Access

Поддерживаются две модели аутентификации — Integrated Windows Authentication и Forms Based Authentication, последняя рекомендуется к использованию совместно с SSL.

Team System Web Access работает на Windows Server 2003, требует наличия на сервере Internet Information Services 6 с включенной поддержкой ASP.NET 2.0 и установленным Microsoft Visual Studio 2005 Team Explorer, на клиенте поддерживаются браузеры Internet Explorer 6+ и Mozilla Firefox 1+.

## Microsoft Visual Studio 2005 Team Foundation Server Power Tool

Microsoft Visual Studio 2005 Team Foundation Server Power Tool (ранее назывался Power Toys) — это набор дополнений и утилит, улучшающих функциональность Team Foundation Server. В данный набор включены две новые пакетные утилиты для разработчиков и три утилиты с графическим интерфейсом — редактор шаблонов процессов, набор настраиваемых политик для операции check in и утилита, позволяющая выполнять модульные тесты.

**Team Foundation Server Power Tool Commands (tfpt.exe)** — это пакетная утилита, расширяющая функциональность Team Foundation Version Control. Некоторые команды, включенные в состав этой утилиты, имеют графический интерфейс. Помимо встроенных в утилиту команд, пользо-

вателям также доступны команды Annotate и Treediff, реализованные в Source Control Explorer (в Visual Studio) и Team Explorer.

В состав Team Foundation Server Power Tool включены следующие команды:

Команда	Описание
Unshelve	Используется для получения и объединения изменений в рамках workspace
Rollback	Используется для отмены изменений, внесенных на уровне Team Foundation Server
Online	Используется для создания изменений (pending edits) для файлов, у которых еще нет таких изменений
GetCS	Команда Get Changeset служит для получения изменений для определенного набора изменений (changeset)
UU	Команда Undo Unchanged служит для отмены операций над неизменными файлами (unchanged files), включая операции добавления, редактирования и удаления
Annotate	Используется для загрузки всех версий указанных файлов и отображения построчной информации о том, кто и когда вносил изменения
Review	Данная команда используется для оптимизации процесса обзора кода путем отказа от использования операций check in и shelving
History	Служит для отображения истории ревизий для одного или более файлов или папок. Опция /followbranches позволяет получить историю для вложенных файлов (file branch)
Workitem	Используется для создания, обновления или просмотра элементов проекта
Query	Используется для запуска запроса к элементу проекта и отображения результатов выполнения этого запроса. Если не указан какой-то конкретный запрос, команда отобразит все элементы проекта, присвоенные данному пользователю
TreeDiff	Служит для визуального представления различий между файлами в двух серверных папках, в папке на сервере и локальной папке или двух локальных папках
Treeclean	Используется для отображения и опционального удаления файлов, которые не управляются системой контроля версий в текущем каталоге и подкаталогах. Эта команда может быть полезна в тех случаях, когда требуется удаление временных файлов из локальной области (workspace) — например файлов, созданных компилятором

**Process Template Editor** — при установке Team Foundation Server Power Tool в Microsoft Visual Studio Team System появляется Process Editor — интегрированное средство для создания новых типов элементов проекта и соответствующих компонентов шаблона процессов. Данная утилита тре-

бует установки компонентов Domain-Specific Language Tools for Visual Studio 2005 Redistributable Components, которые можно найти на сайте MSDN.

**Check-In Policy Pack** — это набор настраиваемых политик для операции check in, расширяющих стандартный набор, включенный в Team Foundation Server. В пакет включены следующие политики:

- Custom Path Policy — политика, предоставляющая механизм для указания маршрута к системе контроля версий, а также маршрутов, на которые распространяется данная политика. Данная политика поддерживает сценарий, когда можно применять различные наборы правил для различных папок системы контроля версий.
- Forbidden Patterns Policy — позволяет указать расширение имени файла или регулярное выражение, которые могут использоваться для указания на то, что определенные файлы не должны включаться в систему контроля версий. Это наиболее полезно для DLL-файлов, ряда файлов, порождаемых в процессе сборки проекта или автоматически создаваемых файлов для Web-сайтов, генерируемых в процессе разработки.
- Changeset Comments Policy — позволяет убедиться в том, что текстовое поле Comments в диалоговой панели Check In не является пустым — если в данном текстовом поле отсутствует информация, файл не может быть включен в операцию check in.
- Work Item Query Policy — позволяет указать запрос, к которому относится тот или иной элемент проекта.

**Test Tools Build Task (TestToolsTask)** позволяет выполнять модульное тестирование простым указанием динамически загружаемых библиотек или указанием шаблона имени файла в файле TfsBuild.proj вместо использования файлов метазаданных тестов (.vsmtdi) для указания тестов, которые должны быть запущены. Данная функциональность аналогична запуску пакетной утилиты mstest.exe с опцией /testcontainer для указания тестового контейнера, все тесты из которого должны быть запущены.

## «Ролевые» версии Visual Studio

В состав Visual Studio 2008 Team Suite входят четыре ролевые версии: для архитекторов — **Architecture Edition**, для разработчиков — **Development Edition**, для разработчиков баз данных — **Database Edition** и для тестировщиков **Test Edition**.

### Architecture Edition — Visual Studio для архитекторов

Продукт Team Architect Edition позволяет архитекторам программного обеспечения и инфраструктуры создавать и описывать все компоненты

сервис-ориентированных приложений. Работа архитекторов построена на основе использования специализированных дизайнеров:

- **Дизайнер распределенных приложений (Distributed Application Designer)** позволяет визуально создавать веб-сервисы и сервис-ориентированные архитектуры, обладает поддержкой расширений веб-сервисов (Web Services Enhancements, WSE) и возможностью проверки архитектуры на соответствие логической инфраструктуре. Включает следующие дизайнеры: Application Connection Designer, Logical Datacenter Designer, System Designer и Deployment Designer.
- **Дизайнер логической инфраструктуры (Logical Infrastructure Designer)** позволяет визуально описывать сетевую инфраструктуру, содержит редактор настроек и ограничений, включает генератор и компилятор System Definition Model (SDM)-модели, средства развертывания.
- **Дизайнер классов (Class Designer)** служит для визуального создания классов и генерации соответствующего кода.

Работая в связке, эти дизайнеры обеспечивают синхронизацию изменений с кодом, визуализацию изменений и т. п. операции. Важно то, что эти средства призваны пошагово решать задачу создания комплексных систем и получать на выходе не только архитектуру, но и требования к инфраструктуре и общую модель системы.

На основе артефактов, созданных архитекторами, разработчики создают и тестируют код приложения. Для этих целей служит версия Team Edition для разработчиков.



В Visual Studio 2008 Team System Architecture Edition появился ряд дополнений и улучшений, которые мы рассмотрим ниже.

- Дизайн систем с использованием нисходящего подхода. Теперь у архитекторов появилась возможность использования нисходящего (top-down) подхода к дизайну приложений с использованием System Designer. Имеется возможность либо начать новый дизайн системы, или продолжить работу над существующим — можно добавлять системы, приложения, точки входа непосредственно в описание системы. Точки входа добавляются прямо на границах описания системы и делегируют свое поведение членам системы.
- Синхронизация точек входа веб-сервисов с WSDL-файлами — теперь можно задавать или обновлять сигнатуры операций существующих веб-сервисов, написанных с использованием .NET, указав соответствующий WSDL-файл.
- Генерация проектов ASP.NET Web Application для приложений, использующих технологию ASP.NET за счет использования шаблона ASP.NET Web Application, который генерирует соответствующий тип проекта для

приложения. Эта возможность доступна только для ASP.NET Web-проектов на основе файловой системы (свойство Project Location Type имеет значение File System). Также имеется возможность ре-инжиниринга ASP.NET Web-приложений.

- Сохранение, импорт и экспорт дополнительных прототипов — у архитекторов появилась возможность сохранять или устанавливать дополнительные прототипы, импортируя их, а не редактируя в реестр. Отметим, что редактирование реестра по-прежнему требуется для установки файлов .sdmdocument для прототипов приложений, созданных средствами System Definition Model (SDM) SDK.

В августе 2007 года была выпущена предварительная версия (Community Technology Preview) пакета утилит — Team Architect Power Tools, расширяющих функциональность Team System Architecture Edition путем добавления средств работы с библиотеками классов на уровне диаграмм приложений и системы. Этот пакет можно бесплатно загрузить с сайта MSDN.

## Development Edition — Visual Studio для разработчиков

Помимо традиционных средств для разработки приложений, обеспечиваемых на уровне функциональности Visual Studio Professional Edition (написание и редактирование кода, компиляция, отладка, развертывание), в версию Team Edition для разработчиков входит ряд существенных дополнений, среди которых:

- **Средства статического анализа кода** выполняют нечто схожее с грамматическим анализом кода. Компилятор выполняет проверку синтаксиса и убеждается в том, что объекты, параметры и другие элементы кода допустимы для данного языка программирования. Средства статического анализа кода проверяют семантику и нарушения безопасности, стиля кодирования, руководств по производительности. Для проверки кода на C/C++ Microsoft предлагает PReFast — средство, работающее с .NET и обычным кодом, а также средство FxCop, которое базируется на наборе рекомендаций .NET Class Design Guidelines.
- **Средства профилирования** для того, чтобы получить максимальную производительность кода, Microsoft предлагает ряд профилировщиков кода, входящих в состав Visual Studio Team System и интегрированных в среду разработки. Доступны два типа профилировщиков:
  - Профилировщик на основе «проб» (Sampling profiler) — отслеживает выполнение приложения и по окончании сообщает данные о производительности.
  - Инструментальный профилировщик (Instrumented profiler) — во время компиляции вставляет в код определенные «метки» и сохраняет данные в хранилище Team System при достижении приложением определенной «метки».

- **Средства модульного тестирования** — модульное тестирование на основе атрибутов стало популярным способом проверки качества кода. В состав Visual Studio Team System входят интегрированные средства создания, слежения и выполнения модульных тестов непосредственно из среды разработчика. Модульное тестирование предполагает, что для каждого метода класса пишется определенный тест. Такой тест проверяет все возможные способы поведения класса (в соответствии со спецификацией) и сообщает об обнаруженных ошибках. Модульные тесты крайне полезны при регрессивном тестировании, т. к. они могут выполняться над каждым классом в проекте для того, чтобы убедиться, что изменения в одном классе не привели к изменениям в других.



В Visual Studio 2008 Team System Development Edition появился ряд дополнений и улучшений, которые мы рассмотрим ниже.

- **Метрики кода (Code Metrics)** — набор программных измерений, позволяющих разработчикам «заглянуть» в создаваемый ими код. Используя метрики кода, разработчики могут понять, какие типы и/или методы должны быть переработаны или дополнительно оттестированы. Помимо этого, команды разработчиков могут идентифицировать потенциальные риски, получать представление о текущем состоянии проекта и отслеживать прогресс в процессе работы над проектом.
- **Средства профилирования**полнились рядом дополнительных возможностей, среди которых:
  - Поддержка Windows Vista — поддерживается полное соответствие требованиям безопасности Windows Vista, доступ к IIS по-прежнему требует прав администратора.
  - Сбор данных о распределении стека — включается на странице свойств Performance Session, отображается в отчете Allocation View.
  - Данные проб на уровне строк кода — включено по умолчанию, отображается в отчетах IP и Line View, информация о строках — в отчете Module.
  - Управление в режиме выполнения — автоматический запуск при запуске средства профилирования, возможность включения и отключения сбора данных из пользовательского интерфейса, запуск с отключенным сбором данных, добавление дополнительных пометок без редактирования кода.
  - Фильтрация аналитических данных — фильтрация по отметкам времени, процессам, потокам и пометкам. Функциональность, схожая с системой запросов в модуле слежения за элементами проекта (опция /filterfile позволяет сохранить фильтр в файле).



- Поиск расхождений (Diff) — возможность управления отображаемыми данными, выбор двух отчетов в Performance Explorer и включение режима сравнения простым щелчком мышью, новый отчет о расхождениях в меню Tools, опция командной строки /diff.
- Поддержка счетчиков производительности Windows — возможность фильтрации данных счетчиков, ассоциация со свойствами сессии подсчета производительности, опция командной строки /wincounter.
- Компактные файлы отчетов — создание файлов отчетов небольшого размера на основе полных отчетов, команда Save Analyzed или опция командной строки /summaryfile.
- Копирование в HTML — возможность копирования отчета и прикрепления его к электронному письму.
- Интеграция с нагрузочным и Web-тестированием — создание нагрузочного или Web-теста, создание сессии измерения производительности прямо из теста.
- Поддержка приложений, использующих технологию Windows Communication Foundation в средствах профилирования.

Следующее издание Visual Studio Team System — это издание, предназначенное для работы с базами данных.

## Database Edition — Visual Studio для разработчиков и администраторов баз данных



Новым компонентом в Visual Studio Team System 2008 является издание, предназначенное для разработчиков и администраторов баз данных — Visual Studio Team System 2008 Database Edition. Это издание было доступно с ноября 2006 года в виде отдельного продукта для Visual Studio 2005 под названием Team Edition for Database Professionals и теперь этот компонент включен в состав Visual Studio Team System 2008.

К основным функциям версии для разработчиков баз данных можно отнести управление изменениями в схеме базы данных, модульное тестирование схемы базы данных, а также возможность полноценной интеграции в весь процесс разработки программного продукта с использованием платформы Visual Studio Team System.

В рамках нового типа проекта (Visual Studio Database Project) у разработчиков появляется возможность импортировать схему базы данных из SQL Server 2000 и SQL Server 2005 (версия SQL Server 2008 будет поддерживаться в следующей версии Visual Studio Team System, которая имеет кодовое название «Rosario») и поместить ее в систему контроля версий. Когда требуется

внедрение изменений в схеме базы данных, новая система поддержки проектов позволяет создать скрипты обновления (update scripts) и предоставляет механизм для выполнения этих скриптов в выбранной базе данных.

Основной концепцией Visual Studio Team System 2008 Database Edition является работа с базами данных в отсоединенном, т. н. оффлайн-режиме. Проект представляет собой логически собранный в коллекцию набор файлов, которые содержат схему базы данных и ее различные представления. Проект может работать с любым SCCI-провайдером (например Team Foundation Server, VSS).

Новый **редактор T-SQL**, входящий в состав Visual Studio, существенно повышает продуктивность при написании кода и обеспечивает поддержку параллельного выполнения запросов и отображение клиентской статистики.

Механизм **Rename Refactoring** позволяет быстро переименовать объекты базы данных, включая все ссылки на эти объекты и все зависимые объекты (схема, генераторы данных, Unit Test, SQL-скрипты и т. п.). Обычно такие изменения называют «каскадными обновлениями».

Механизм **Schema Compare** позволяет быстро сравнить схемы двух баз данных (или одной базы данных — в системе контроля версий и в самой базе данных) и создать скрипты обновления для синхронизации схем. Схожий с описанным выше механизм Data Compare позволяет быстро сравнить данные в двух базах данных и создать скрипты обновления для синхронизации данных. Оба механизма генерации скриптов поддерживают выполнение как непосредственно из среды Visual Studio, так и из командной строки (SQLCMD), SQL Server Management Studio или в рамках исполняемого файла.

Механизм **Data Generator** служит для создания плана генерации повторяющихся наборов «осмысленных» данных на основе реальных данных, взятых из базы данных с поддержкой гистограмм и распределений. Такие данные используются для заполнения базы при выполнении модульных тестов или нагрузочного тестирования. Механизм генерации данных поддерживает модули расширения — т. о. есть возможность создания и подключения собственных генераторов.

И, наконец, поддержка **модульного тестирования базы данных** (Database Unit Testing) на основе Team Test Unit Test позволяет создавать тесты, используя язык T-SQL или управляемый код (Visual Basic .NET или C#). Можно тестировать следующие объекты баз данных: хранимые процедуры, функции, триггеры, любой код на языке T-SQL. Также имеется возможность тестирования базы данных на уровне всего разрабатываемого приложения.

### Ролевой подход

При работе с Visual Studio Team System 2008 Database Edition предполагается использование двух ролей — администратора базы данных и разра-

ботчика базы данных. Функции по фазам работы над проектом — администрирование, разработка, внедрение распределяются следующим образом.

На первом этапе — администрировании — администратор базы данных выполняет следующие действия:

- Создает в Visual Studio новый проект базы данных.
- Получает схему из базы данных.
- Создает план генерации данных.

Затем наступает этап разработки, на котором разработчик базы данных:

- Разрабатывает тесты.
- Пишет код базы данных и объектов.
- Выполняет рефакторинг.
- Запускает тесты.
- Вносит код и тесты в систему управления версиями.

На финальной стадии — внедрении — снова задействуется администратор базы данных, который:

- Проверяет внесенные изменения.
- Сравнивает модификации с рабочей базой данных.
- Создает скрипты для внедрения (скрипты для изменения схемы базы данных или самих данных).
- Внедряет скрипты в рабочую базу данных.

Выпущенный в июле 2007 пакет обновлений — **Team Edition for Database Professionals SR1**, в котором реализована поддержка ссылок на базы данных в рамках проекта (cross-database reference) и ссылок на метафайлы баз данных (.dbmeta). Помимо этого улучшена поддержка файлов в файловых группах SQL Server на уровне свойств проекта и к свойствам базы данных добавлена страница Variables, на которой можно задавать переменные типа setvar, используемые в скриптах развертывания. И, наконец, данный пакет обновлений полностью поддерживает пакет обновлений для SQL Server 2005 — Service Pack 2 и работу под управлением операционной системы Windows Vista.

## Test Edition — Visual Studio для тестировщиков

В состав издания Visual Studio Team System для тестировщиков входят средства для нагрузочного тестирования и измерения производительности Web-приложений. Важно отметить то, что средства нагрузочного тестирования входят непосредственно в средство разработки и тесно интегрированы как со средой, так и с механизмами сбора данных и создания отчетов.

### Запись и воспроизведение тестов

Такой подход позволяет обеспечить максимально тесное взаимодействие разработчиков и тестировщиков — при использовании Visual Studio 2005

Team System больше нет необходимости в обмене списком ошибок и последующим слежением за их исправлением — все данные (описания ошибок с необходимым контекстом) доступны из одной среды и соответствующие работы (Work Items) назначаются непосредственным исполнителям.

В Visual Studio Team System появилась возможность записи действий тестировщиков при работе с веб-приложениями и последующего воспроизведения таких записей. Такие записанные тесты хранятся в виде XML-файлов — они могут быть сохранены в системе контроля версий вместе с кодом приложения и другими артефактами периода разработки.

### **Нагрузочное тестирование**

После того как тест веб-приложения записан и сохранен, он может быть воспроизведен на инфраструктуре веб-фермы с соответствующими нагрузочными данными. Условия загрузки настраиваются с помощью мастера а Load Test Creation Wizard и сохраняются в виде XML-файлов — они также могут быть сохранены в системе контроля версий вместе с кодом приложения и другими артефактами периода разработки.

Все нагрузочные тесты создаются и выполняются непосредственно из среды разработки Visual Studio. Это позволяет тестировщикам и разработчикам обмениваться информацией, используя одно и то же средство, доступное любой роли.

Тестировщики могут просматривать результаты тестов и создавать отчеты об ошибках на основе данных, полученных при нагрузочном тестировании. Такие отчеты содержат поля для разработчиков, позволяющие в точности повторить обнаруженную проблему, а также информацию, собранную в процессе выполнения приложения. Таким образом, разработчики не только видят отчет, но и получают возможность проанализировать данные, вызвавшие ошибку.

Помимо этого, как мы отметили выше, Visual Studio Team System поддерживает механизмы модульного тестирования, которое может выполняться и в нагруженном состоянии.

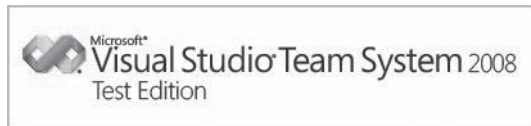
### **Управление тестированием**

Все тесты, создаваемые в Visual Studio Team System, включая модульные тесты, воспроизводимые веб-тесты и нагрузочные тесты, управляются из окна Test Manager среды разработчика. Это позволяет разработчикам, тестировщикам и менеджерам продуктов получать информацию о статусе и полноте выполнения всех тестов для данного программного проекта. Помимо этого, все тесты могут быть сохранены в систем контроля версий Visual Studio Team Foundation Server.

Окно Test Manager позволяет создавать так называемые «списки тестов», которые полезны в том случае, когда из всего набора тестов для данного программного проекта при получении новой версии необходимо выпол-

нить только часть тестов. Поддерживается возможность отображения тестов по «владельцам», типам и другим характеристикам.

В Visual Studio 2008 Team System Test Edition появился ряд дополнений и улучшений, которые мы рассмотрим ниже.

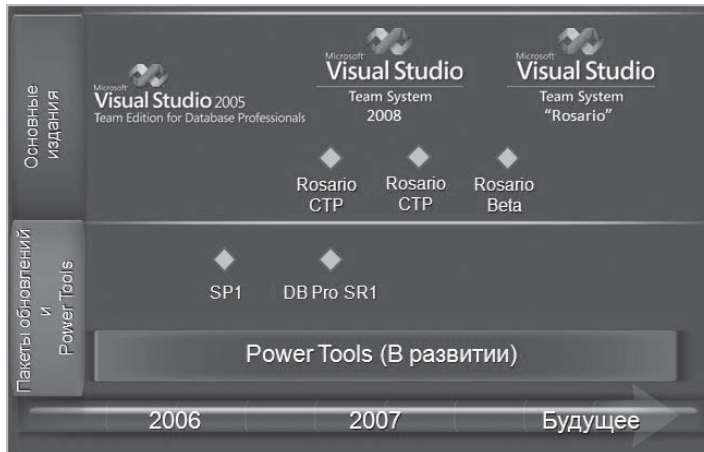


- Возможность создания и выполнения модульного тестирования в Visual Studio Professional Edition.
- Упрощенное выполнение модульных тестов — поддерживается возможность запуска тестов непосредственно из кода.
- Использование наследования между тестовыми классами.
- Выполнение модульного тестирования на устройствах, разработка для которых поддерживается в рамках Visual Studio.
- Создание новых хост-адаптеров, позволяющих выполнять тесты в средах, отличных от сред тестирования по умолчанию.
- Улучшенная связь с источниками данных для модульного тестирования, включая CSV-файлы, XML-файлы и т. д.
- Улучшенная связь с источниками данных в веб-тестах.
- Улучшенные правила проверки результатов тестов при веб-тестировании — более гибкая система применения правил и использования результатов для управления выполнением веб-тестов.
- Более гибкое управление нагрузочным тестированием и моделированием нагрузок.
- Улучшенные средства просмотра результатов нагрузочного тестирования, включая новые диаграммы и представления.
- Улучшенные средства управления репозитарием результатов нагрузочного тестирования.
- Сохранение результатов тестирования в виде XML-файлов со схемой. Возможность программной обработки результатов тестирования, автоматически сохраняемых в XML-формате — файлы с расширением .trx (Test Result XML).

## **Visual Studio Team System в развитии — проект «Rosario»**

Относительно недавно компания Microsoft раскрыла планы по развитию семейства продуктов Visual Studio Team System и предоставила доступ к

ранней версии следующего поколения продуктов этого семейства, известного под кодовым названием «Rosario».



#### Планы по развитию Visual Studio Team System

Microsoft Visual Studio Team System «Rosario» представляет собой интегрированное средство для управления всем жизненным циклом приложения (Application Life-cycle Management, ALM) и состоит из инструментов, процессов и руководств по их применению. При разработке новой версии Visual Studio Team System Microsoft решает две ключевых бизнес-задачи: синхронизация разработки приложений с требованиями бизнеса и улучшение качества приложений и их тестирования. К средствам, поддерживающим решение первой задачи, относятся:

- Средства планирования портфеля проектов — интеграция с Portfolio Server через Project Server.
- Проектное планирование — интеграция с Project Server и упрощенные средства управления расписаниями.
- Средства сбора требований и их проверки.
- Отслеживание требований за счет поддержки иерархических связей и улучшенной системы запросов.
- Улучшенная система генерации отчетов.

Для решения второй задачи — улучшение качества приложений и их тестирования — планируются следующие улучшения в продукте:

- Управление сценариями тестирования.
- Выполнение тестов — как функциональных, так и ручных.
- Интеграция средств тестирования и диагностики для облегчения воспроизведения ошибок.
- Приоритет тестов — возможность определения наиболее важных для выполнения тестов.

Часть из планируемых расширений и улучшений уже включена в предварительную версию продукта. Их использование позволит проектной команде:

- Более эффективно общаться между собой и заказчиками, а также выполнять совместные действия в рамках работы над программным проектом.
- Обеспечивать ожидаемое качество разрабатываемого программного обеспечения, используя соответствующие средства на каждом шаге создания продукта.
- Получать детальную информацию об активностях в рамках проекта и приоритетах на основе данных, которые позволят всем членам команды принимать обоснованные решения.

В версии, выпущенной для предварительного ознакомления в августе 2007 года, реализована поддержка нескольких сценариев работы над программным проектом:

- Создание расписания разработки и его отслеживание. Проектные команды могут более эффективно обмениваться задачами и отчетами о прогрессе в рамках проекта, используя механизмы слежения за элементами проекта, включая возможность создания дочерних соотношений между элементами проекта, определения зависимостей задач и отслеживания прогресса работы над требованиями (от определения до реализации) через актуальные отчеты и таблицы. Это возможно либо через Visual Studio Team Explorer, либо за счет более тесной интеграции с Microsoft Excel
- Тестирование требований и ручное тестирование. Возможность определения соотношения между сценариями тестирования и требованиями к программному продукту дает тестировщикам возможность обнаружить пропущенные сценарии еще на ранних стадиях проекта, а также более точно определить причины неудачного выполнения тестовых сценариев. При выполнении ручных тестов, используя утилиты Manual Test Runner, позволяющей выполнять тестирование вне интегрированной среды, тестировщикам предоставляется возможность включения в отчеты большей информации об обнаруженных ошибках. С помощью Manual Test Runner можно загружать тесты с сервера Team Foundation Server, публиковать результаты тестирования, включать в отчеты копии экранов, автоматически собирать системную информацию, показывать шаги для воспроизведения ошибки, а также записывать видеоролики, показывающие эти шаги.
- Управление зависимостями. Разработчикам предоставляется возможность более простого определения зависимостей, которые основываются на нескольких задачах и, таким образом, упростить расстановку приоритетов для всех членов проектной группы.

Рассмотренные выше сценарии создания расписания разработки и отслеживания этого расписания, тестирования требований и ручного те-

стирования, а также управления зависимостями показывают ряд направлений развития Visual Studio Team System «Rosario». Новая функциональность позволяет более быстро создавать качественные приложения любого класса, более эффективно работать в команде вне зависимости от сложности проекта или размера проектной группы.

## Механизмы расширения Visual Studio

Ниже мы кратко рассмотрим основные механизмы, используемые для расширения функциональности среды разработчика Microsoft Visual Studio. Интегрированная среда разработчика поддерживает различные способы создания расширений — от простых макросов, которые могут быть связаны либо с определенными комбинациями клавиш, либо с кнопками на панели инструментов, и дополнительных модулей (Add-Ins), позволяющих добавлять к среде новую функциональность, до механизмов VS Package, используя которые можно встраивать в среду разработчика целые программные продукты или существенно изменять или расширять функциональность среды.

Ниже мы рассмотрим каждый из трех доступных механизмов расширения функциональности интегрированной среды разработчика Visual Studio.

### Макросы

Использование макросов — самый простой способ создания расширений для Visual Studio. Запись макроса позволяет сохранить последовательность использования практически всех команд среды и комбинаций клавиш. Помимо этого, механизм макросов дает вам доступ к объектной модели управления Visual Studio (Visual Studio Automation Object Model), содержащей более 140 различных объектов, а также доступ к классам .NET Framework. Несмотря на свою гибкость, макросы имеют ряд ограничений — для их написания используется язык Visual Basic, они не могут использоваться для создания новых окон (Tool Window), команд или страниц настройки опций (Tools Options).

### Дополнительные модули

Создание дополнительных модулей (Add-Ins) — это второй механизм расширения функциональности Visual Studio. Дополнительные модули предоставляют возможность полного управления Visual Studio через Automation Object Model, что существенно облегчает взаимодействие с большинством инструментов и функций среды разработчика, включая Text Editor, Output Window, Task List и Code Model. Для создания модулей расширения в Visual Studio входит специальный «мастер» — Add-In Solution Wizard.

В отличие от макросов, модули расширения могут быть написаны на любом языке программирования, поддерживающем технологию COM. К таким языкам, в частности, относятся Visual C#, VB.Net и Visual C++. Среди



других отличий модулей расширения от макросов отметим то, что модули сохраняются в откомпилированных DLL, что обеспечивает большую гибкость при распространении таких модулей, а также тот факт, что модули расширения могут использоваться для создания новых команд, страниц опций и окон инструментов (Tool Window). С другой стороны, используя модули расширения нельзя создавать новые типы документов, новые типы проектов, новые средства отладки и т. п. — для решения этих задач используются т. н. VSIP-пакеты.

## Механизмы VS Package

Механизмы VSIP-пакетов обеспечивают максимальную гибкость при решении задач, связанных с расширением функциональности среды разработчика в Visual Studio. VSIP-пакеты обеспечивают доступ к ряду программных интерфейсов (т. н. VSIP-интерфейсы), используя которые можно получить практически безграничный контроль над средой разработчика и интегрировать в Visual Studio практически любое программное обеспечение. К возможностям, предоставляемым VSIP-интерфейсами, относятся создание новых типов проектов, новых отладчиков, редакторов, расширений дизайнеров и новых языков программирования. Одним из примеров использования интерфейсов VSIP для интеграции нового языка программирования является IronPython — подробности интеграции описаны в Visual Studio SDK.

Начиная с Visual Studio 2005 у разработчиков появилась возможность использования библиотеки Managed Package Framework для более простого создания VSIP-пакетов на управляемых языках программирования.

## Расширения Visual Studio Team System

Помимо рассмотренных выше механизмов, которые доступны во всех версиях Visual Studio за исключением версии Express, в Visual Studio Team System существует ряд дополнительных возможностей. Среди них — возможность расширения шаблонов процессов, расширение «мастера» создания проектов, расширение типов элементов работы (Work Items), расширение типов тестов, настройки системы построения проектов (Build System), а также управление системой контроля версий через Version Control Object Model и расширение механизма формирования отчетов.

## Visual Studio 2008 Shell



Visual Studio 2008 Shell — это максимально «облегченная» версия Visual Studio, которую можно бесплатно использовать для создания собственных

средств разработки. Visual Studio 2008 Shell предоставляет основу, на базе которой можно реализовывать различные дизайнеры, средства проектирования, разработки, собственные языки программирования — пользователи таких средств будут работать в уже знакомой им по опыту использования Visual Studio среде.

Среди преимуществ использования Visual Studio Shell отметим быструю разработку за счет наличия среды, позволяющей включать в нее дополнительные инструментальные средства и языки программирования, знакомую среду программирования, использование которой существенно сокращает время на освоение и изучение, а также оптимизацию для хостинга языков программирования и различных программных инструментов.

Visual Studio Shell можно использовать в двух режимах — интегрированном и изолированном. В первом режиме расширения, созданные для Visual Studio Shell, объединяются (интегрируются) с любым другим изданием Visual Studio, установленным на компьютере, во втором — приложения работают параллельно с другими изданиями Visual Studio и изолированы от них. Интегрированный режим представляет интерес, в первую очередь, для компаний, создающих средства, расширяющие функциональность Visual Studio, тогда как изолированный режим — это отличное средство для компаний, создающих собственные инструменты.

Visual Studio 2008 Shell доступна в составе VS 2008 SDK, который можно загрузить с сайта MSDN. Более подробную информацию о механизмах расширения Visual Studio, включая Visual Studio 2008 Shell, можно получить в блоге [http://blogs.msdn.com/vsxteam/](http://blogs.msdn.com/vsxtteam/).

## **Visual Studio Team System. Полезные ссылки**

### **Общие ресурсы**

- Российский раздел сайта Microsoft Developer Network (MSDN)
  - <http://msdn2.microsoft.com/ru-ru/default.aspx>
- Раздел сайта MSDN, посвященный Visual Studio Team System
  - <http://msdn2.microsoft.com/teamsystem/>

### **Team Foundation Server**

- Team Foundation Server Team Center
  - <http://msdn2.microsoft.com/en-us/teamsystem/aa718934>
- Руководство «Team Development with Visual Studio Team Foundation»
  - <http://www.codeplex.com/tfsguide>

- Visual Studio Team Foundation Server — Project Server 2007 Connector
  - <http://www.codeplex.com/pstfsconnector>
- Microsoft Visual Studio 2005 Team Foundation Server Power Tool
  - <http://go.microsoft.com/?linkid=5422499>
- Project Manager Team Center
  - <http://msdn2.microsoft.com/en-us/teamsystem/aa718827>
- Ответы на часто задаваемые вопросы по Visual Studio Team System
  - <http://msdn2.microsoft.com/en-us/teamsystem/aa718930.aspx>

### **Для архитекторов**

- Software Architect Team Center
  - <http://msdn2.microsoft.com/en-us/teamsystem/aa718756>
- Ответы на часто задаваемые вопросы по Team Edition for Software Architects
  - <http://msdn2.microsoft.com/en-us/teamsystem/aa718909.aspx>

### **Для разработчиков**

- Software Developer Team Center
  - <http://msdn2.microsoft.com/en-us/teamsystem/aa718765>
- Ответы на часто задаваемые вопросы по Visual Studio 2005 Team Edition for Software Developers
  - <http://msdn2.microsoft.com/en-us/teamsystem/aa718914.aspx>
- Class Designer PowerToys
  - <http://www.gotdotnet.com/workspaces/workspace.aspx?id=fe72608b-2b28-4cc1-9866-ea6f805f45f3>

### **Для разработчиков и администраторов баз данных**

- Database Developers Team Center
  - <http://msdn2.microsoft.com/en-us/teamsystem/aa718764>
- Ответы на часто задаваемые вопросы по Visual Studio Team Edition for Database Professionals
  - <http://msdn2.microsoft.com/en-us/teamsystem/aa718913.aspx>
- Visual Studio 2005 Team Edition for Database Professionals Power Tools
  - <http://go.microsoft.com/?linkid=7244111>

### **Для тестировщиков**

- Software Tester Team Center
  - <http://msdn2.microsoft.com/en-us/teamsystem/aa718941>
- Ответы на часто задаваемые вопросы по Visual Studio 2005 Team Edition for Software Testers

- <http://msdn2.microsoft.com/en-us/teamsystem/aa718931.aspx>
- patterns & practices Performance Testing Guidance
  - <http://www.codeplex.com/perftesting>

## Visual Studio 2008

- <http://msdn2.microsoft.com/en-us/vstudio/aa700830.aspx>

## Visual Studio «Rosario»

- <http://msdn2.microsoft.com/en-us/teamsystem/bb725993.aspx>

## Механизмы расширения Visual Studio

- Visual Studio Extensibility Developer Center
  - <http://msdn2.microsoft.com/en-us/vstudio/aa700819.aspx>
- VSX Team Blog
  - <http://blogs.msdn.com/vsxteam/>
- Domain-Specific Language Tools Developer Center
  - <http://msdn2.microsoft.com/en-us/vstudio/aa718368.aspx>
- Visual Studio Industry Partner (VSIP) Program
  - <http://msdn2.microsoft.com/en-us/vstudio/aa700860.aspx>
- Visual Studio 2008 Shell
  - <http://msdn2.microsoft.com/en-us/vstudio/bb510103>
  - <http://msdn.microsoft.com/vstudio/shell/>

## Партнерские решения

- Teamlook — организация эффективной командной работы на основе Outlook <http://www.personifydesign.com/products/teamlook/teamlook.aspx>
- TeamSpec — управление требованиями <http://www.personifydesign.com/products/teamspec/teamspec.aspx>
- Интеграция с Eclipse
  - <http://www.teamprise.com/>
- Бизнес моделирование
  - <http://www.ravenflow.com/>
- Project Server 2007 VSTS Connector
  - <http://www.codeplex.com/pstfsconnector>
- Requirements Authoring Starter Kit — решение по управлению требованиями на основе TFS в исходных кодах <http://www.microsoft.com/downloads/details.aspx?FamilyID=E96CCC54-8759-452F-BF68-3A261C663B66&displaylang=en>

- Шаблон процесса SCRUM для TFS
  - <http://www.scrumforteamssystem.com/en/default.aspx>

## Приложение. Аппаратные и программные требования

Издание Microsoft Visual Studio 2008 Team System Team Foundation Server представляет собой многозвенное приложение, компоненты которого могут быть установлены на различных компьютерах. Для установки Visual Studio Team Foundation Server на одном компьютере вам потребуется машина со следующими характеристиками:

- Процессор с частотой не ниже 2.2 ГГц
- Минимум 2 Гбайт оперативной памяти
- 8 Гбайт свободного пространства на жестком диске со скоростью не менее 5400 об/мин
- Дисплей с разрешением не менее 1024 x 768
- Привод DVD

Пакет Visual Studio Team Foundation Server может быть установлен на следующие операционные системы:

- Windows Vista (x86 и x64) — все издания за исключением издания Starter Edition
- Windows XP (x86 и x64) с установленным пакетом обновлений Service Pack 2 или более поздней версии — все издания за исключением издания Starter Edition
- Windows Server 2003 (x86 и x64) с установленным пакетом обновлений Service Pack 1 или более поздней версии
- Windows Server 2003 R2 (x86 или x64) или более поздней версии
- Windows Server 2008 (x86 и x64)

Ниже мы более подробно рассмотрим аппаратные и программные требования для каждого компонента Visual Studio Team System.

Компонент	Программные требования	Аппаратные требования
Клиентские компоненты	32-битная операционная система: <ul style="list-style-type: none"> <li>• Windows XP Professional/Service Pack 2</li> <li>• Windows Server 2003</li> <li>• Windows Vista</li> </ul> 64-битная операционная система: <ul style="list-style-type: none"> <li>• Microsoft Windows Server 2003/SP1, Standard x64 Edition (режим WOW)</li> <li>• Microsoft Windows Server 2003/SP1, Enterprise x64 Edition (режим WOW)</li> </ul>	Минимум: процессор: 2.0 GHz, 512 MB памяти, 8 GB HDD. Рекомендуется: процессор: 2.6 GHz, 1 GB памяти, 20 GB HDD

Компонент	Программные требования	Аппаратные требования
TFS Слой данных	<ul style="list-style-type: none"> <li>• Microsoft Windows Server 2003/SP1, Datacenter x64 Edition (режим WOW)</li> <li>• Microsoft Windows Server 2003 R2, Standard x64 Edition (режим WOW)</li> <li>• Microsoft Windows Server 2003 R2, Enterprise x64 Edition (режим WOW)</li> <li>• Microsoft Windows Server 2003 R2, Datacenter x64 Edition (режим WOW)</li> <li>• Microsoft Windows XP Professional x64 Edition (режим WOW)</li> <li>• Windows Vista</li> </ul> <p>Требуемые программные компоненты</p> <ul style="list-style-type: none"> <li>• Internet Explorer 6.0/ Service Pack 1</li> <li>• Microsoft Office 2003/Service Pack 1 или</li> <li>• Microsoft Office 2007</li> <li>• MDAC 9.0</li> <li>• .NET Framework 2.0</li> </ul> <p>32-битная операционная система:</p> <ul style="list-style-type: none"> <li>• Windows Server 2003, Standard Edition/ Service Pack 1 (SP1)</li> <li>• Microsoft Windows Server 2003 R2 Standard Edition</li> <li>• Windows Server 2003, Enterprise Edition/ Service Pack 1 (SP1)</li> <li>• Microsoft Windows Server 2003 R2 Enterprise Edition</li> <li>• Windows Server 2003 Datacenter Edition/ Service Pack 1 (SP1)</li> <li>• Microsoft Windows Server 2003 R2 Datacenter Edition</li> <li>• Windows Server 2008</li> </ul> <p>TFS Слой данных не поддерживает 64-битные операционные системы при развертывании на одном сервере с прикладным слоем</p> <p>64-битная операционная система:</p> <ul style="list-style-type: none"> <li>• Microsoft Windows Server 2003/Service Pack 1 (SP1), Standard x64 Edition</li> <li>• Microsoft Windows Server 2003/Service Pack 1 (SP1), Enterprise x64 Edition</li> <li>• Microsoft Windows Server 2003/Service Pack 1 (SP1), Datacenter x64 Edition</li> <li>• Microsoft Windows Server 2003 R2, Standard x64 Edition</li> </ul>	<p>Минимум: процессор: 2.2 GHz, 8 GB HDD, 1 GB памяти.</p> <p>Сервер должен обладать надежным сетевым соединением с минимальной скоростью 1 Mbps и максимальной задержкой в 350 ms.</p> <p>Рекомендованная конфигурация: см. следующий раздел</p>

Компонент	Программные требования	Аппаратные требования
TFS Прикладной слой	<ul style="list-style-type: none"> <li>• Microsoft Windows Server 2003 R2, Enterprise x64 Edition</li> <li>Microsoft Windows Server 2003 R2, Datacenter x64 Edition</li> </ul> 32-битная операционная система: <ul style="list-style-type: none"> <li>• Windows Server 2003 Standard Edition/ Service Pack 1 (SP1)</li> <li>• Microsoft Windows Server 2003 R2 Standard Edition</li> <li>• Windows Server 2003 Enterprise Edition/ Service Pack 1 (SP1)</li> <li>• Microsoft Windows Server 2003 R2 Enterprise Edition</li> <li>• Windows Server 2003 Datacenter Edition/ Service Pack 1 (SP1)</li> <li>• Microsoft Windows Server 2003 R2 Datacenter Edition</li> <li>• Windows Server 2008</li> </ul> TFS Прикладной слой не поддерживает 64-битные операционные системы	Минимум: процессор: 2.2 GHz, 8 GB HDD, 1 GB памяти. Сервер должен обладать надежным сетевым соединением с минимальной скоростью 1 Мbps и максимальной задержкой в 350 ms. Рекомендованная конфигурация: см. следующий раздел
Team Build	32-битная операционная система: <ul style="list-style-type: none"> <li>• Windows Server 2003/ Service Pack 1 (SP1), Service Pack 2 (SP2)</li> <li>• Windows Server 2003 R2</li> <li>• Windows XP Professional, .NET 2.0</li> </ul> 64-битная операционная система: <ul style="list-style-type: none"> <li>• Microsoft Windows Server 2003/SP1, Standard x64 Edition (режим WOW)</li> <li>• Microsoft Windows Server 2003/SP1, Enterprise x64 Edition (режим WOW)</li> <li>• Microsoft Windows Server 2003/SP1, Datacenter x64 Edition (режим WOW)</li> <li>• Microsoft Windows Server 2003 R2, Standard x64 Edition (режим WOW)</li> <li>• Microsoft Windows Server 2003 R2, Enterprise x64 Edition (режим WOW)</li> <li>• Microsoft Windows Server 2003 R2, Datacenter x64 Edition (режим WOW)</li> </ul> Microsoft Windows XP Professional x64 Edition (режим WOW)	Минимум: процессор: 766 GHz, 8GB HDD, 256MB памяти. Рекомендованная конфигурация: см. следующий раздел
TFS Proxy	32-битная операционная система: <ul style="list-style-type: none"> <li>• Windows Server 2003 Standard Edition/ Service Pack 1 (SP1)</li> <li>• Microsoft Windows Server 2003 R2 Standard Edition</li> <li>• Windows Server 2003 Enterprise Edition/Service Pack 1 (SP1)</li> </ul>	Минимум: процессор: 2 GHz, 40 GB HDD, 2 GB памяти. Сервер должен обладать надежным сетевым соединением с минимальной скоростью 1 Мbps и максимальной задержкой в 350 ms.

Компонент	Программные требования	Аппаратные требования
Удаленное тестирование	<ul style="list-style-type: none"> <li>• Microsoft Windows Server 2003 R2 Enterprise Edition</li> <li>• Windows Server 2003 Datacenter Edition/Service Pack 1 (SP1)</li> <li>• Microsoft Windows Server 2003 R2 Datacenter Edition</li> <li>• Windows Server 2008</li> </ul> TFS Proxy не поддерживает 64-битные операционные системы Windows Server 2003/ Service Pack 1 (SP1), Windows XP Professional/ Service Pack 2 или Windows 2000/Service Pack 4, SQL Server 2005 Express Edition, .NET Framework 2.0	Процессор: 600 MHz, 1 GB HDD, 512 MB памяти. Рекомендованная конфигурация: см. следующий раздел

## Рекомендованные аппаратные конфигурации

### Team Foundation Server. Прикладной слой

Конфигурация	CPU	HD	Память	Комментарии
<b>Менее 20 пользователей</b>	2.2 GHz	8 GB	1 GB	Установка на том же компьютере, что и слой данных (один сервер)
<b>От 20 до 50 пользователей</b>	2.2 GHz	30 GB	1 GB	Установка на том же компьютере, что и слой данных (один сервер)
<b>От 50 до 100 пользователей</b>	3.4 GHz	30 GB	1 GB	Установка на том же компьютере, что и слой данных (один сервер)
<b>От 100 до 200 пользователей</b>	3.4 GHz	40 GB	2 GB	Установка на том же компьютере, что и слой данных (один сервер)
<b>От 200 до 400 пользователей</b>	3.4 GHz/ Dual	80 GB	2 GB	Установка на том же компьютере, что и слой данных (один сервер)
<b>От 400 до 800 пользователей</b>	2.8 GHz	20 GB	1 GB	Установка на отдельном от слоя данных компьютере (два сервера)
<b>От 800 до 2000 пользователей</b>	2.8 GHz/ Dual	40 GB	4 GB	Установка на отдельном от слоя данных компьютере (два сервера)

### Team Foundation Server. Слой данных

Конфигурация	CPU	HD	Память	Комментарии
<b>Менее 20 пользователей</b>	2.2 GHz	8 GB	1 GB	Установка на том же компьютере, что и прикладной (один сервер)
<b>От 20 до 50 пользователей</b>	2.2 GHz	30 GB	1 GB	Установка на том же компьютере, что и прикладной (один сервер)
<b>От 50 до 100 пользователей</b>	3.4 GHz	30 GB	1 GB	Установка на том же компьютере, что и прикладной (один сервер)



Конфигурация	CPU	HD	Память	Комментарии
От 100 до 200 пользователей	3.4 GHz	40 GB	2 GB	Установка на том же компьютере, что и прикладной (один сервер)
От 200 до 400 пользователей	3.4 GHz/ Dual	80 GB	2 GB	Установка на том же компьютере, что и прикладной (один сервер)
От 400 до 800 пользователей	2.8 GHz/ Dual	80 GB	2 GB	Установка на отдельном от прикладного слоя компьютере (два сервера)
От 800 до 2000 пользователей	2.8 GHz/ Quadruple	150 GB	4 GB	Установка на отдельном от прикладного слоя компьютере (два сервера)

### Team Build. Рекомендованные аппаратные конфигурации

Конфигурация	Время сборки	CPU	HD	Память
Минимальная, 1-2 проекта, 5-20 пользователей	30 мин.	766 GHz	8 GB	256 MB
Небольшая, 2-20 проектов, 20-100 пользователей	От 30 мин. до 2 часов	1.5 GHz	30 GB	512 MB
Средняя, более 20 проектов, от 100 до 250 пользователей	От 2 до 5 часов	2.6 GHz	50 GB	1 GB
Большая, более 50 проектов, от 250 до 500 пользователей	От 3 до 7 часов	2.8 GHz/ Dual	80 GB	2 GB

### Конфигурации для тестирования на удаленных компьютерах

Конфигурация	Компонент	CPU	HD	Память
Минимальная, 1-2 проекта, 5-20 пользователей	test agent	600 GHz	1 GB	256 MB
Минимальная, 1-2 проекта, 5-20 пользователей	test controller	600 MHz	1 GB	256 MB
Минимальная, 1-2 проекта, 5-20 пользователей	оба	600 MHz	1 GB	512 MB
Небольшая, 2-20 проекта, 20-100 пользователей	test agent	2.0 GHz	5 GB	512 MB
Небольшая, 2-20 проекта, 20-100 пользователей	test controller	1.0 GHz	8 GB	512 MB
Небольшая, 2-20 проекта, 20-100 пользователей	Оба компонента	2.0 GHz	8 GB	1 GB
Средняя, более 20 проектов, от 100 до 250 пользователей	test agent	2.6 GHz	5 GB	2 GB

Подпишись  
на бюллетень  
MSDN

Узнай новости  
первым!



Бюллетень MSDN  
Новости разработчиков  
[msdn.microsoft.com/ru-ru/flash](http://msdn.microsoft.com/ru-ru/flash)