

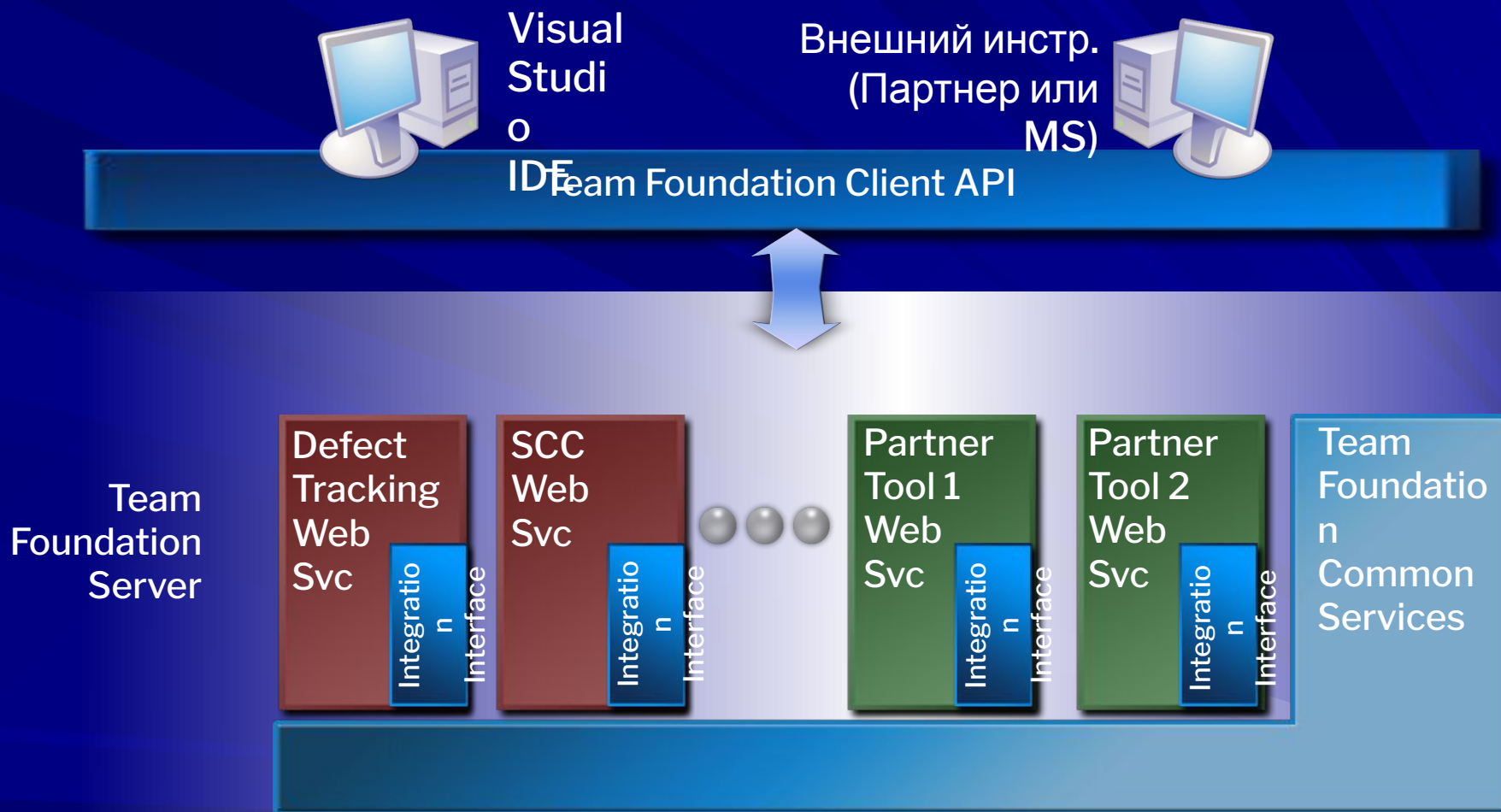
# Team Foundation Server

Основы Расширяемости

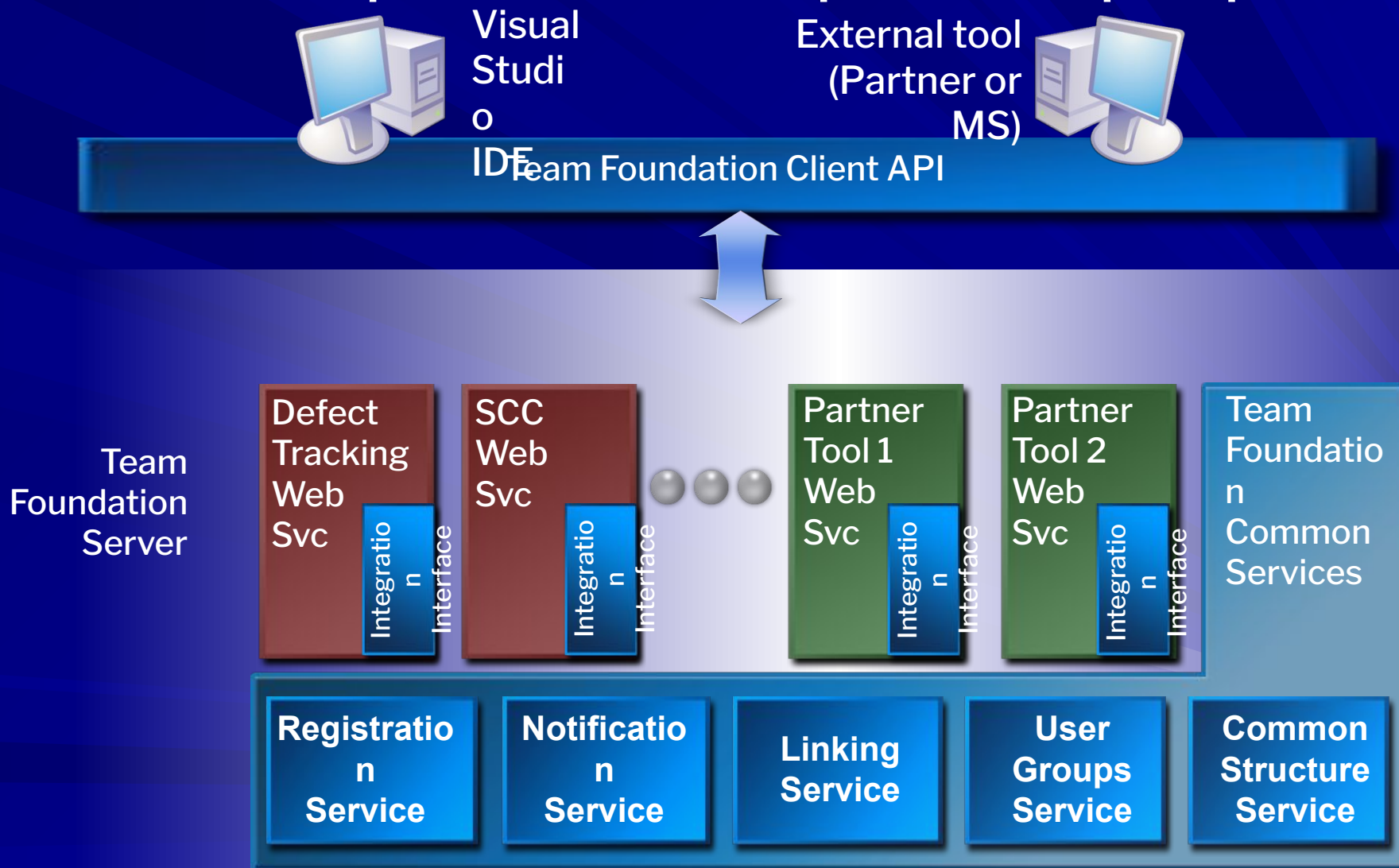
# Девиз Расширяемости TFS

Для интеграции в/с среду TFS,  
ВЫ можете использовать то  
же самое, что и МЫ

# Основные службы интеграции



# Интеграция со стороны сервера



# Концептуально

TFS Server	Один Team Foundation Server и набор интегрированных инструментов. «Логический сервер» может быть на разных формах хранения
Контекст Проекта	артефактов, групп пользователей, и т.д. Все инструменты TFS распознают
Артефакт	Набор данных, который требует отслеживания.
Тип Артефакта	Тип набора данных, который требует отслеживания.
URI Артефакта	Уникальный неизменяемый идентификатор артефакта.

# Настройки и расширения

- Настройки
  - Что? Подгонка TFS к вашей среде
  - Кто? Пользователи, менеджеры проекта
  - Зачем? Настроить под нужды проекта
  - Как? Через GUI , XML, автоматизация...
- Расширения/Интеграция:
  - Что? Новая функциональность
  - Кто? Партнеры, внутренняя IT
  - Зачем? Встроить новое в платформу
  - Как? Веб-службы, БД, VS SDK

# Примеры настроек

- Шаблон проекта
  - Встроенные – изм.
  - Новые - создать
- Типы Work Item
  - Встроенные- изм.
  - Новые – создать
  - Автоматизация (WIOM)
- Верс. контроль
  - Выбрать политики
- Отчеты
  - Изм. встроенные
  - Новые – создать
  - Использовать другой тул для отчетов (Excel, компоненты Office Web)
- Портал Проекта
  - Настраиваемый шаблон
  - Новые библиотеки

# Примеры расширений

- Основные службы
  - Новый тул с новыми артефактами, линками, БД, веб-службами
  - Линки к другим артефактам
  - Ответ на события
  - Интеграция с Team Explorer
  - Новые страницы в Мастере создания проекта
- Версионный Контроль:
  - Об'ектная Модель
  - Поддержка новых типов файлов (diff)
  - Интегрированный checkin
  - Новые политики
  - Ответ на события
- Отчеты
  - Расширения warehouse



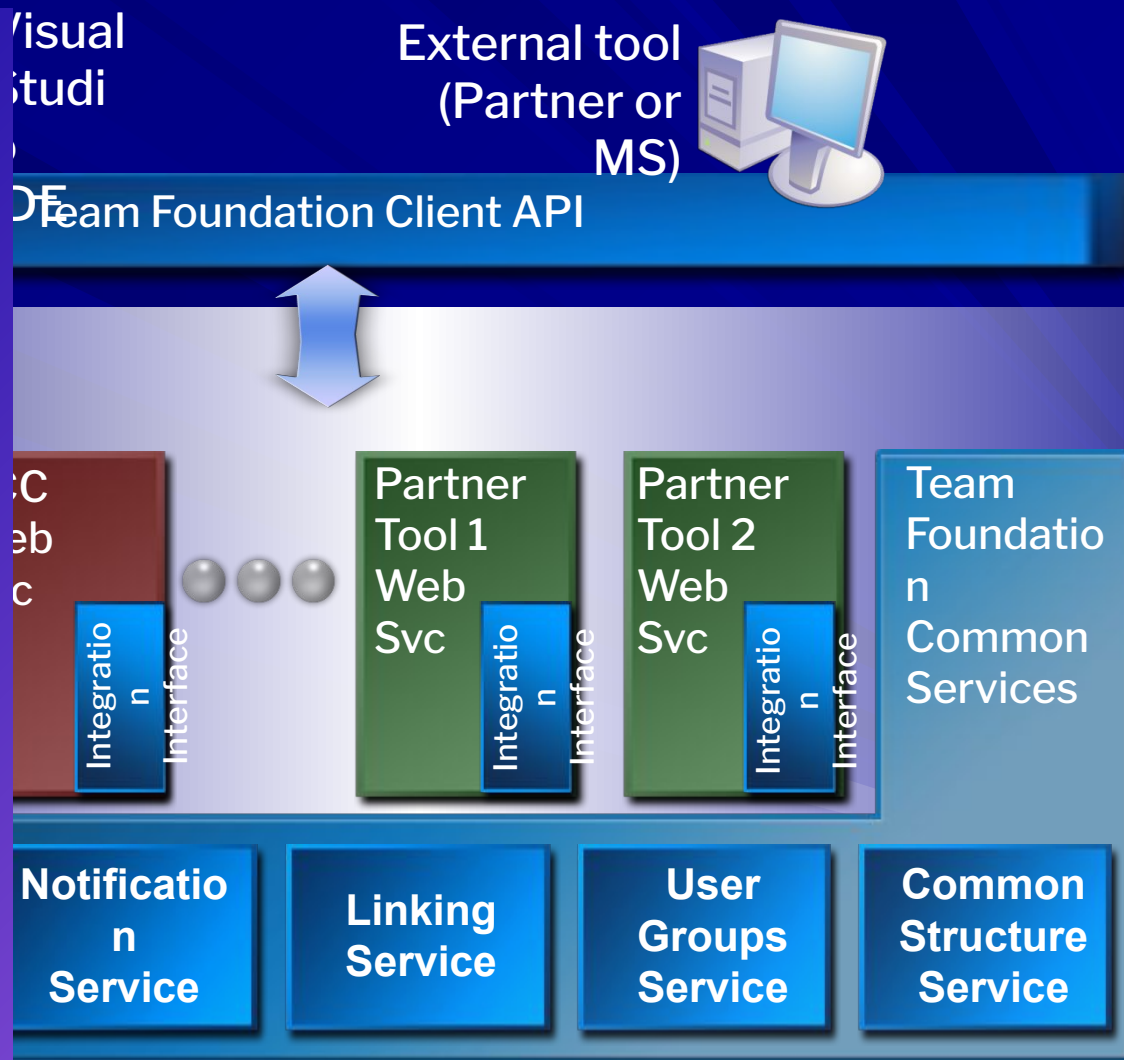
# Примеры расширений

- Work Item Tracking
  - Об'ектная Модель
  - Ответ на события
  - Связь с WI
  - Интеграция с WI picker
- SPS Портал
  - Новые web parts
- Билд
  - Новые задачи билда (build tasks)

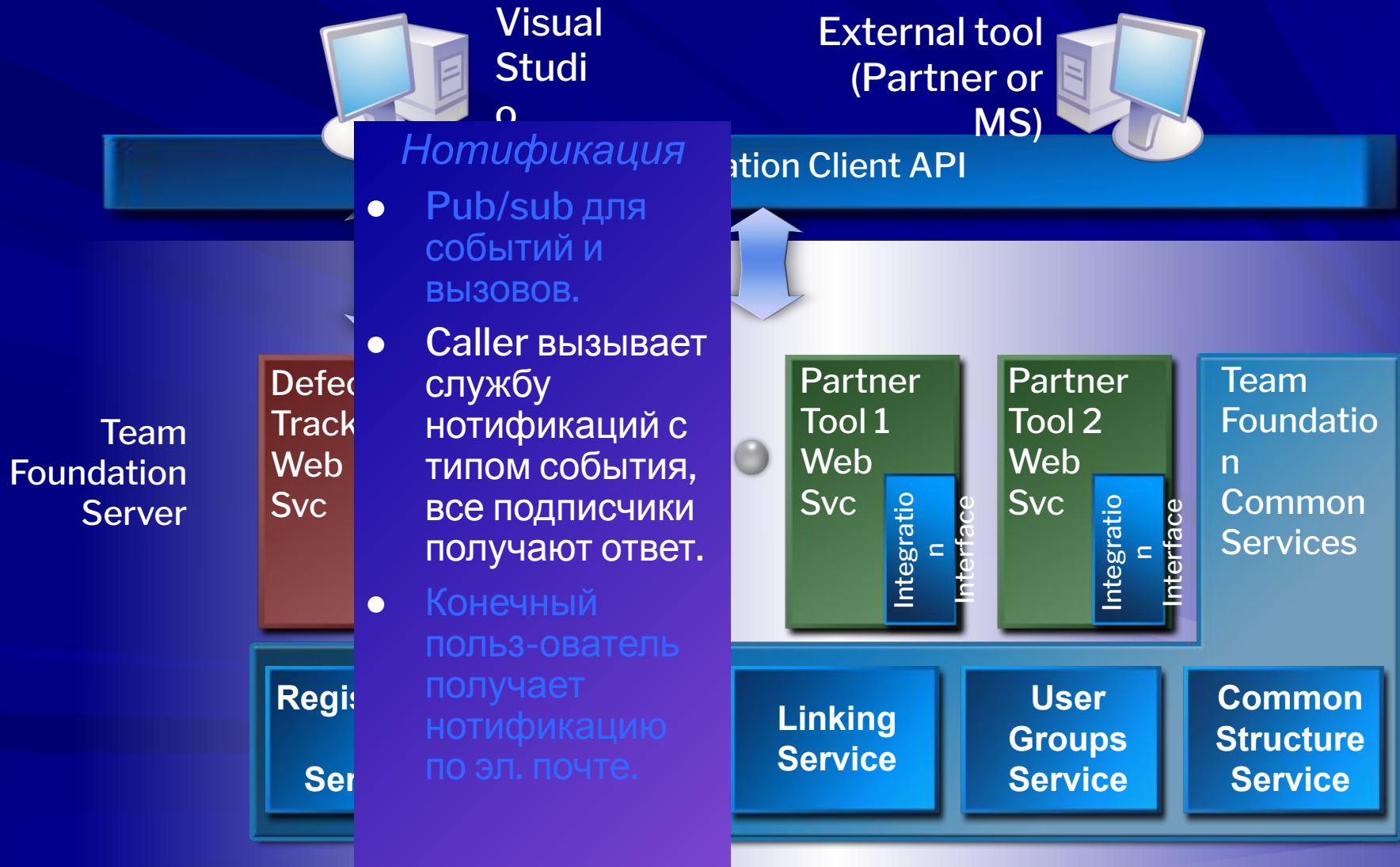
# Регистрация

## Регистрация

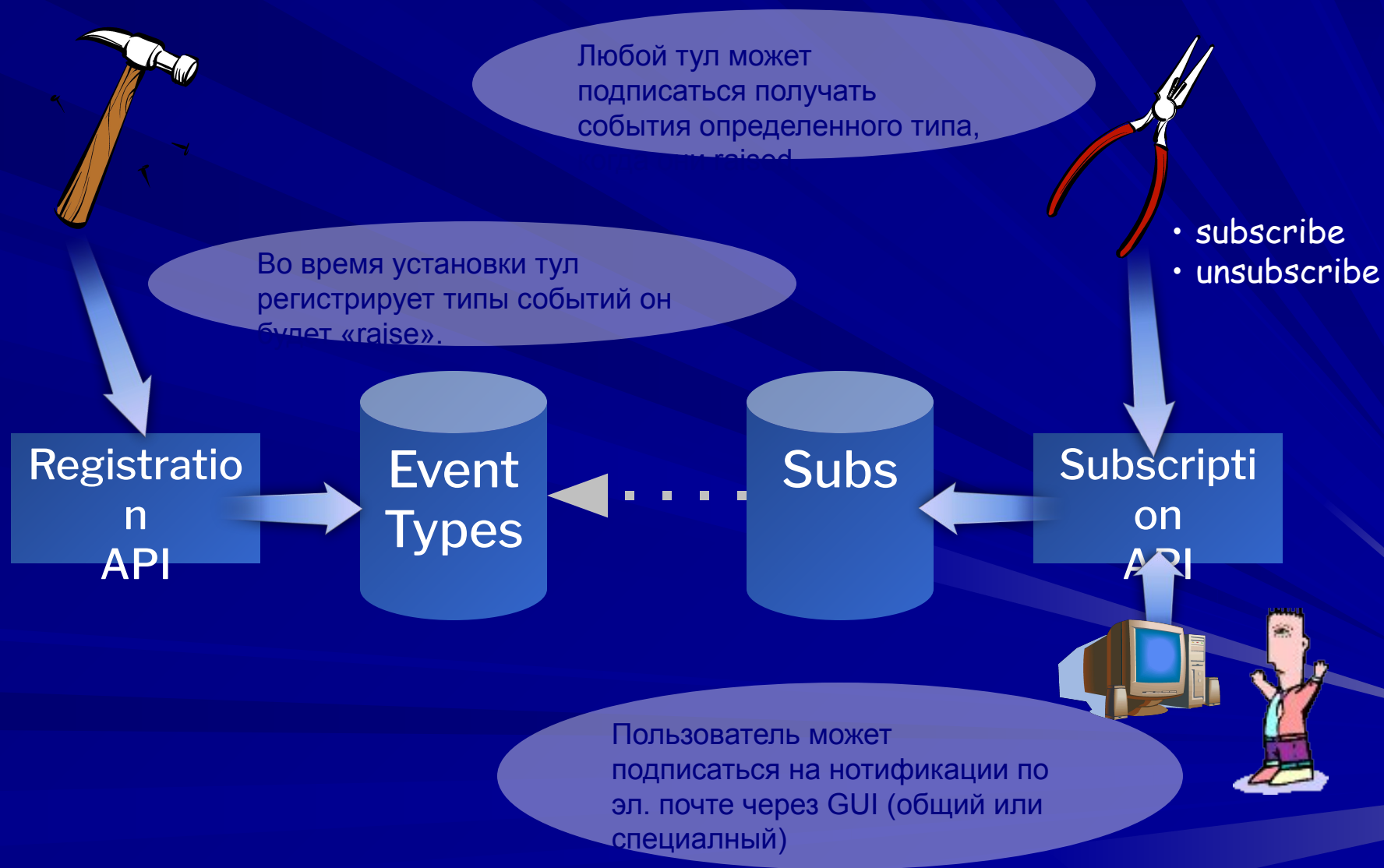
- Направляет callers к службам/инстр. Caller должен знать только URL веб-службы TFS location table.
- Если серверы поменялись – единственное место для изменений, одно для всех тулов, VS IDE и т.д.
- **Location table** может кэшироваться для быстрого действия и обновляться по необходимости
- Тулы регистрируют типы событий, артефактов, линков во время установки



# Нотификации

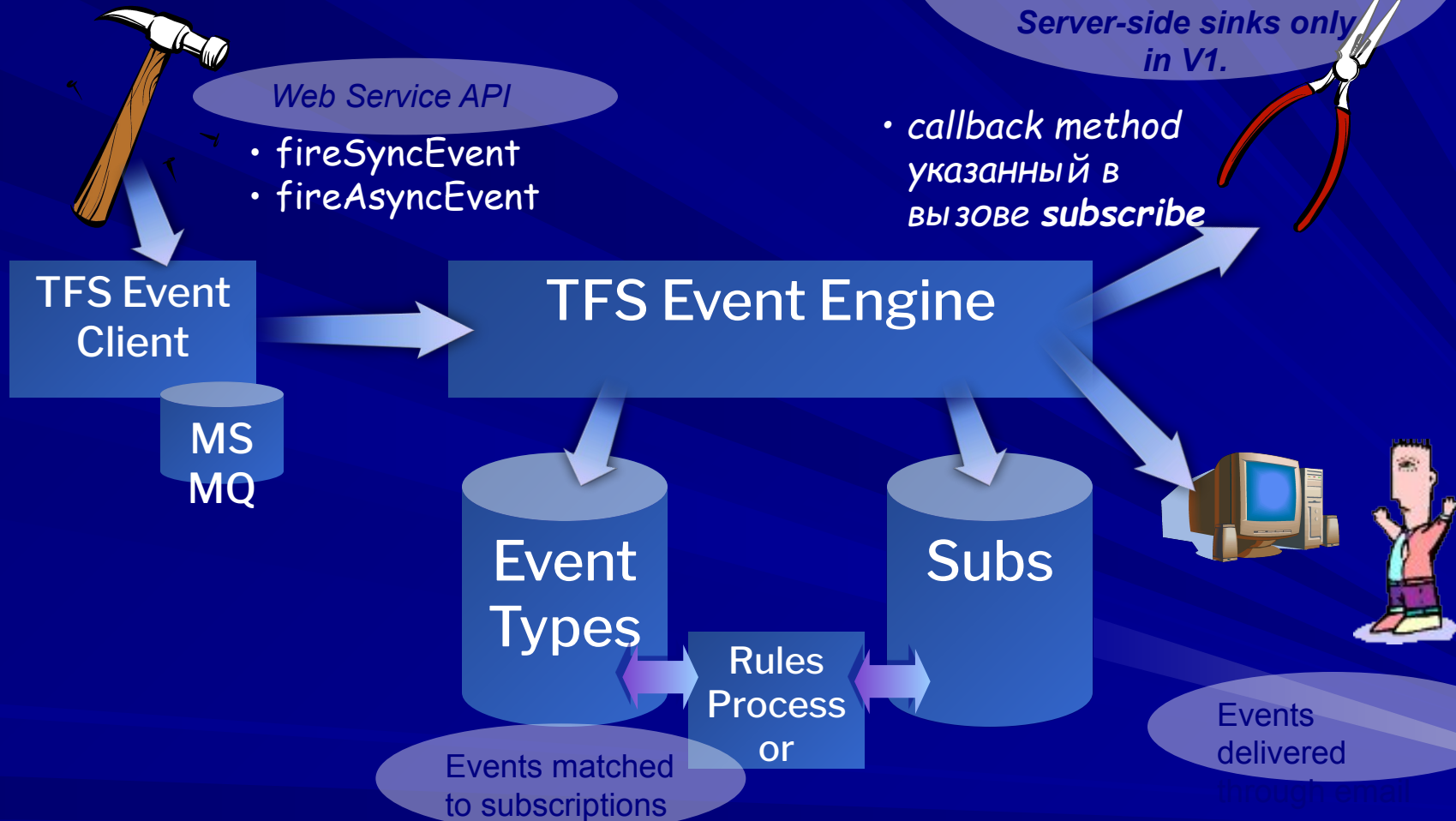


# Регистрация и Подписка



# Raising Событие

События приходят  
через web service  
callback. **Note:**  
*Server-side sinks only  
in V1.*



- Reliable delivery on request
- Event order is preserved

# Обзор API Нотификаций

- FireSyncEvent метод
- FireAsyncEvent метод
- Subscribe метод
- Unsubscribe метод
- Subscriptions метод (*возвращает список всех подписок*)
- *Callout для нотификаций указываемый во время подписки.*



# Linking



Visual  
Studi

o

IDE Team Founda



External tool  
*Linking*

- Слабосвязанные ссылки между данными тулов -URI-based СВЯЗИ.
- Полные и частичные инверсионные запросы, кто на меня указывает?
- Тулы обязаны предоставлять URIs для СВЯЗЫВАЕМЫХ объектов.

Team  
Foundation  
Server

Defect  
Tracking  
Web  
Svc

Integratio  
n  
Interface

SCC  
Web  
Svc

Integratio  
n  
Interface

Registratio  
n  
Service

Notificatio  
n  
Service

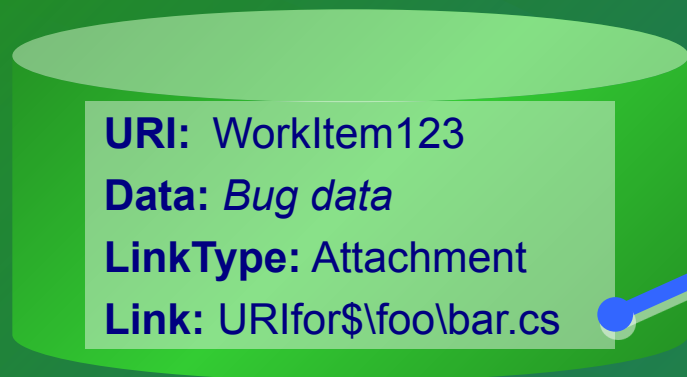
Service

Team  
Foundatio  
n  
Common  
Services

Common  
Structure  
Service

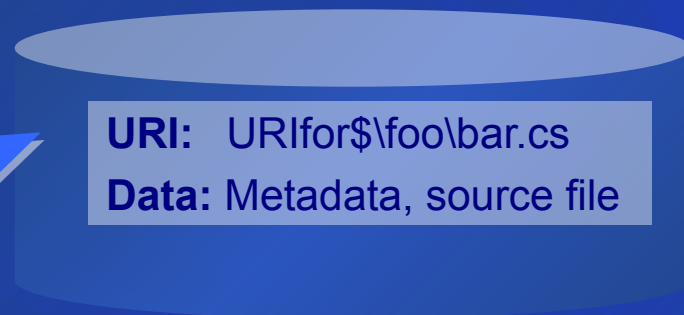
# Пример

## Work Item Tracking



- Имеет URI к артефактам управляемых SCC включая:
- Поддержку
  - *GetReferencingArtifacts*
  - *GetArtifacts*
  - *Execute DefaultAction*
- Линки на другие об'екты (опция)

## Source Code Control



- Создает и показывает для каждого артефакта SCC включая:
  - File
    - Folder
    - ChangeSet
- Поддерживает
  - *GetArtifacts*
  - *ExecuteDefaultAction*



# Основы Linking

- Artifacts - Артефакты
- Artifact Identifiers (URIs):

`http://TfsServer/vset/servertype.instance/artifacttype/moniker`

- Links – Линки, Связи
- Таргет (target) линка не должна ничего знать о *sources*, которые на нее указывают.
- Для более «серьезных» дел, поддерживается также и tight coupling.

# Пример GUI связывания Linking UI

Тип линка

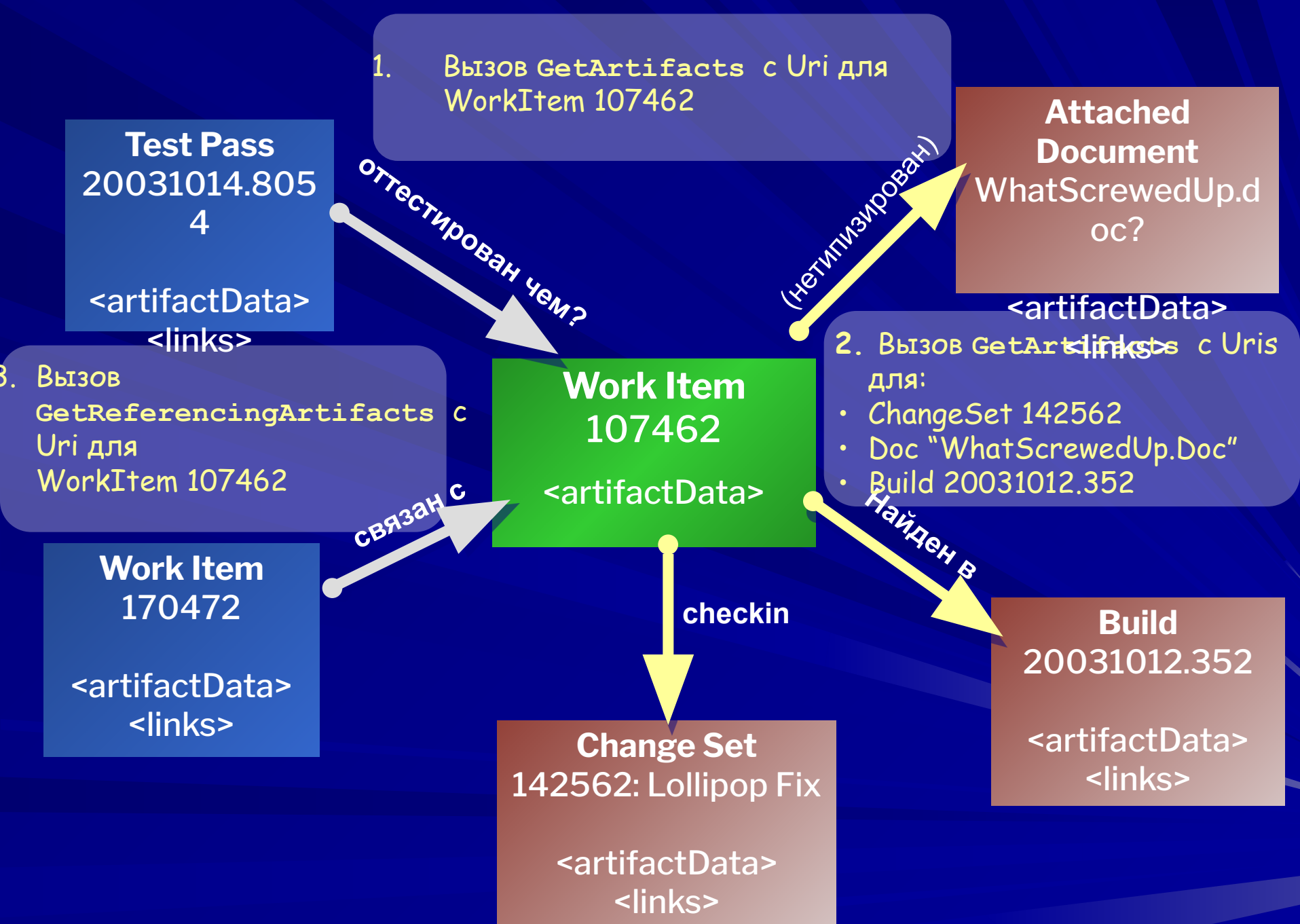
Легкое имя  
Friendly Name

**Work Item 107462 участвует в таких связях:**

Relationship	Item type	Item	Modified By		
involves	attachment	WhatScrewedUp.doc	10/13/2003	10:00am	pm1
found in	build	20031012.352	10/12/2003	9:52pm	qa1
checkin	change set	142562: Lollipop fix	10/15/2003	6:42pm	dev1
tested by	test pass	20031014.8054	10/14/2003	2:15pm	
related to	Work Item	170472	10/5/2003	3:15pm	

Тип  
Артефакта

Последнее  
изменение  
данных по  
отношению к  
общей схеме



1. Call `GetArtifacts` with Uri for WorkItem 107462

Test Pass  
20031014.805  
4

<artifactData>  
<links>

tested by

(untyped)

Attached  
Document  
WhatScrewedUp.d  
oc?

<artifactData>  
<links>

2. Call `GetArtifacts` with Uri for

### Work Item 107462 участвует в таких связях:

Relationship	Item type	Item	Modified	By
involves	attachment	WhatScrewedUp.doc	10/13/2003 10:00am	pm1
found in	build	20031012.352	10/12/2003 9:52pm	qa1
checkin	change set	142562: Lollipop fix	10/15/2003 6:42pm	dev1
тестирован	test pass	20031014.8054	10/14/2003 2:15am	qa1
связан с	Work Item	170472	10/5/2003 3:13pm	gematthew

<artifactData>  
<links>

Change Set  
142562: Lollipop Fix

<artifactData>  
<links>

<artifactData>  
<links>

# Линк и тип линка

- Линк (*link*) может иметь тип
- Тип линка описывает назначение/цель линка
- Для отображения, тип линка имеет 2 роли (reading):
  - consumer *linkreading1* provider
  - provider *linkreading2* consumer
  - Пример:
    - Тест *x* тестирует Метод *y*
    - Метод *y* тестируется Тестом *x*
- Действующие типы линков регистрируются тулами во время установки.
- Типы линков могут также указывать с какими типами артефактов они работают (если известно заранее)

# Внешняя адресация

- Вы можете предоставить страницу ASP для показа ваших артефактов.
- URI артефактов встроенные в emails, и т.д., и попадают на ваши ASP страницы через TFS ISAPI фильтр.
- Примечания:
  - Страницы хранятся в определенном virtual directory.
  - XSL для форматирования данных артефактов.
  - Сложная ASP.Net application.
  - Нет ASP страницы? Пользователь получит 404.

`http://VisualStudioServer8/vset/vsversionstore.2/csharpfile/b4e321`

6

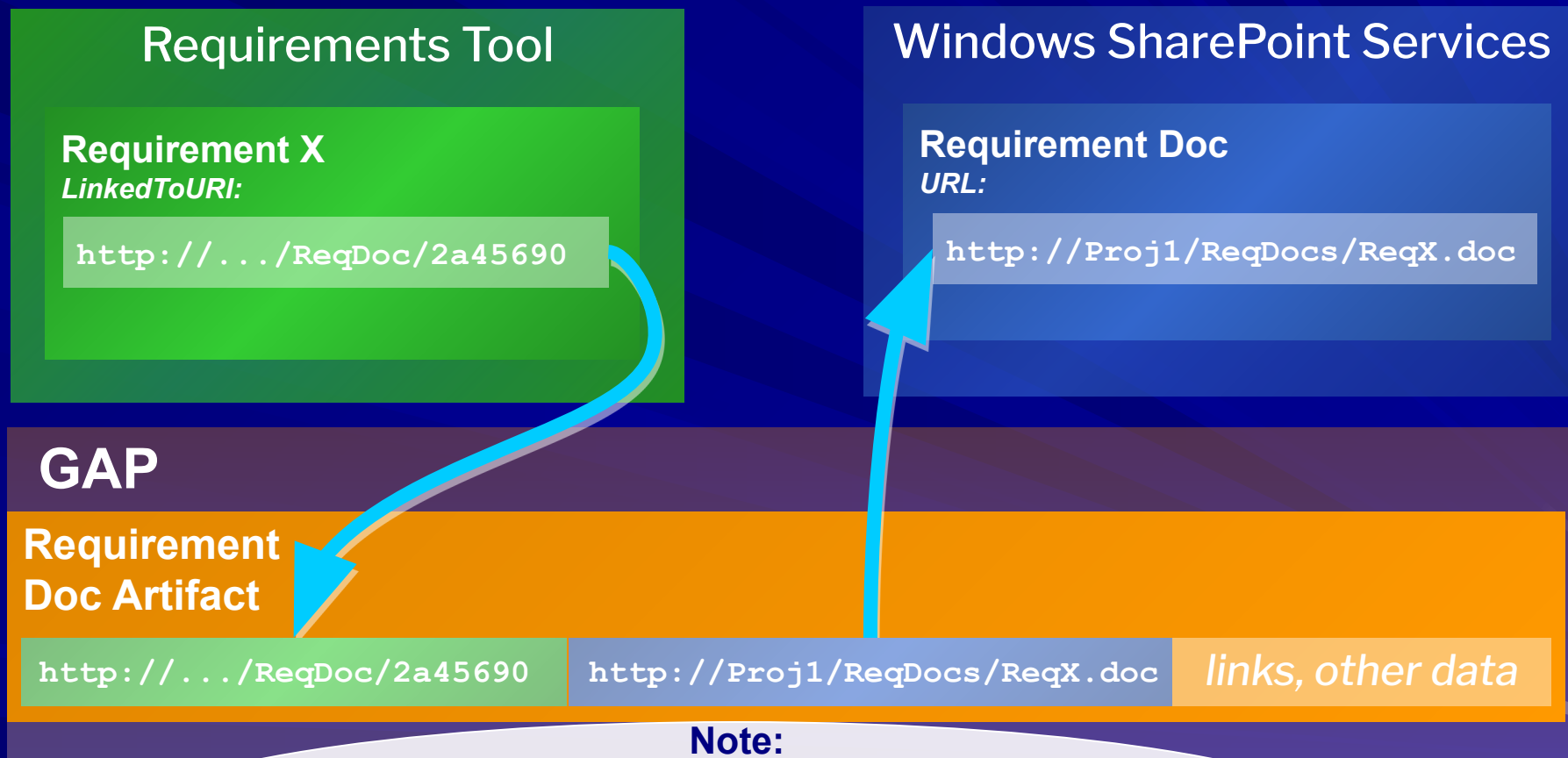
*становится...*

`http://VisualStudioServer8/artasp/vsversionstore.2.asp?type=csharpfile,id=b4e321`

6

# Generic Artifact Provider (GAP)

## Оболочка для инструментов вне TFS

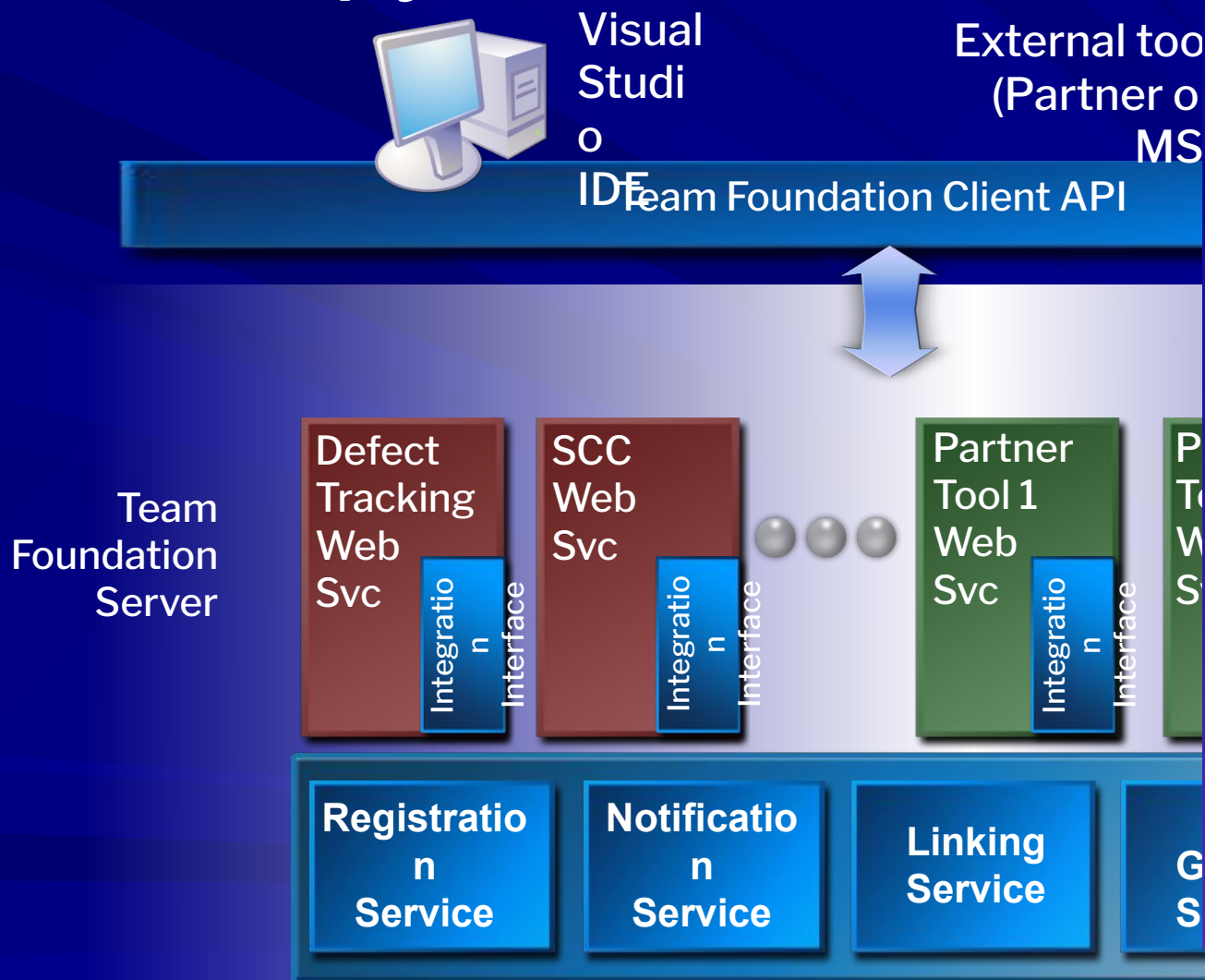


### Note:

GAP Adapter нужен для каждого инструмента.  
Adapter должен получать сообщения об  
изменениях в артефактах



# Группы Пользователей



## User Groups

- Lightweight user groups / roles for security and policy.
- Default groups for common roles.
- Group resolution includes AD groups and TFS groups.
- TFS Enabled Tools will include these user groups when they perform authorization.
- In addition, TFS provides simple auth services for tools that can use them



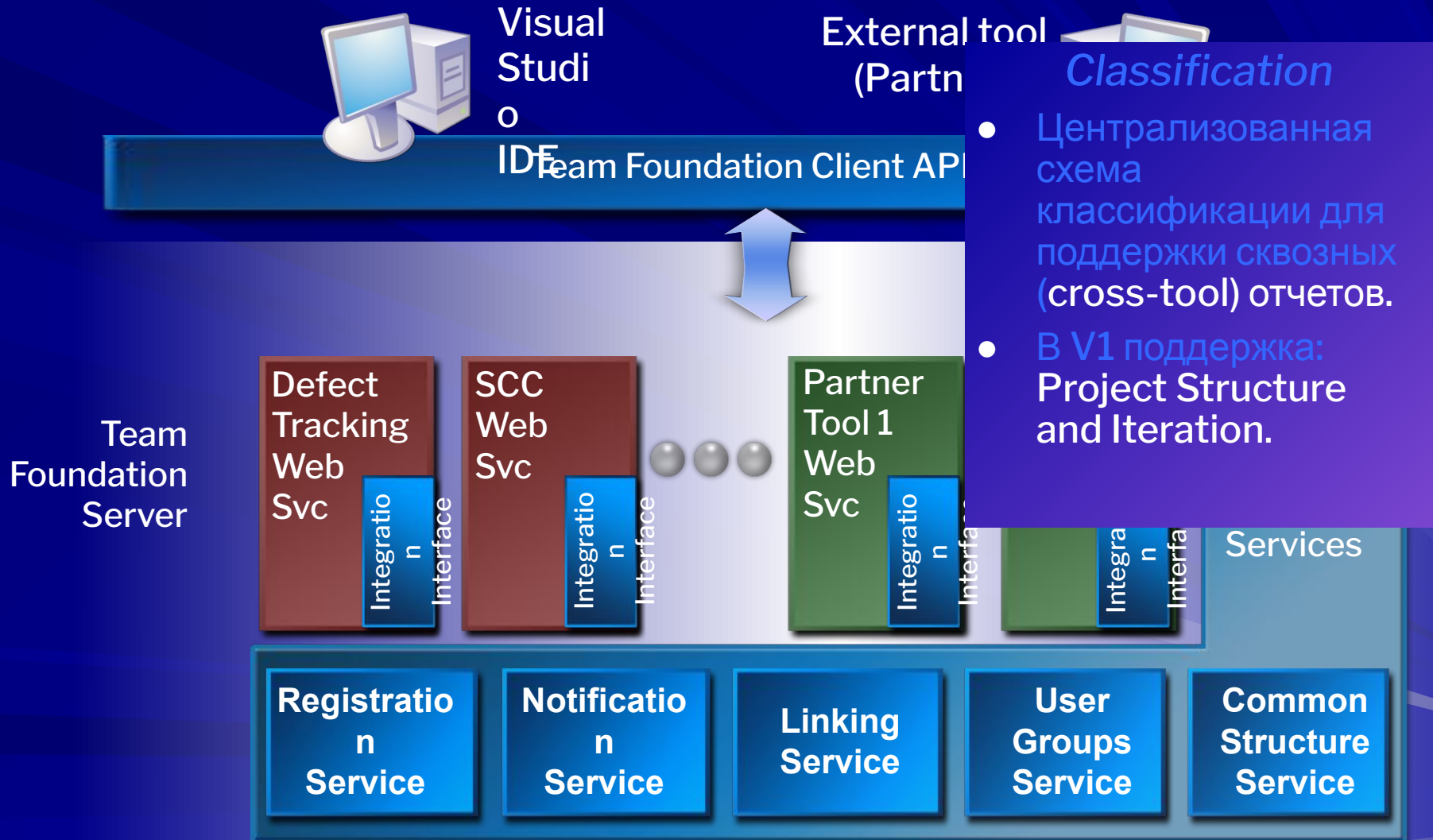
# User Group Служба

- Требует компоненты Win 2003 Server
  - Authorization Manager (AzMan)
  - Active Directory Application Mode (ADAM)
- Пользователи могут определять собственные группы
- По определению, тулы с поддержкой TFS знают о преопределенных группах **Administrators** и **Project Administrators**.
- Определения групп могут передаваться в WSS, SQL Reporting Services.
- Для авторизации:
  - Пишите свои собственные и пользуйтесь группами TFS *or*
  - Пользуйтесь нашим механизмом AzMan авторизации

# Обзор User Group API

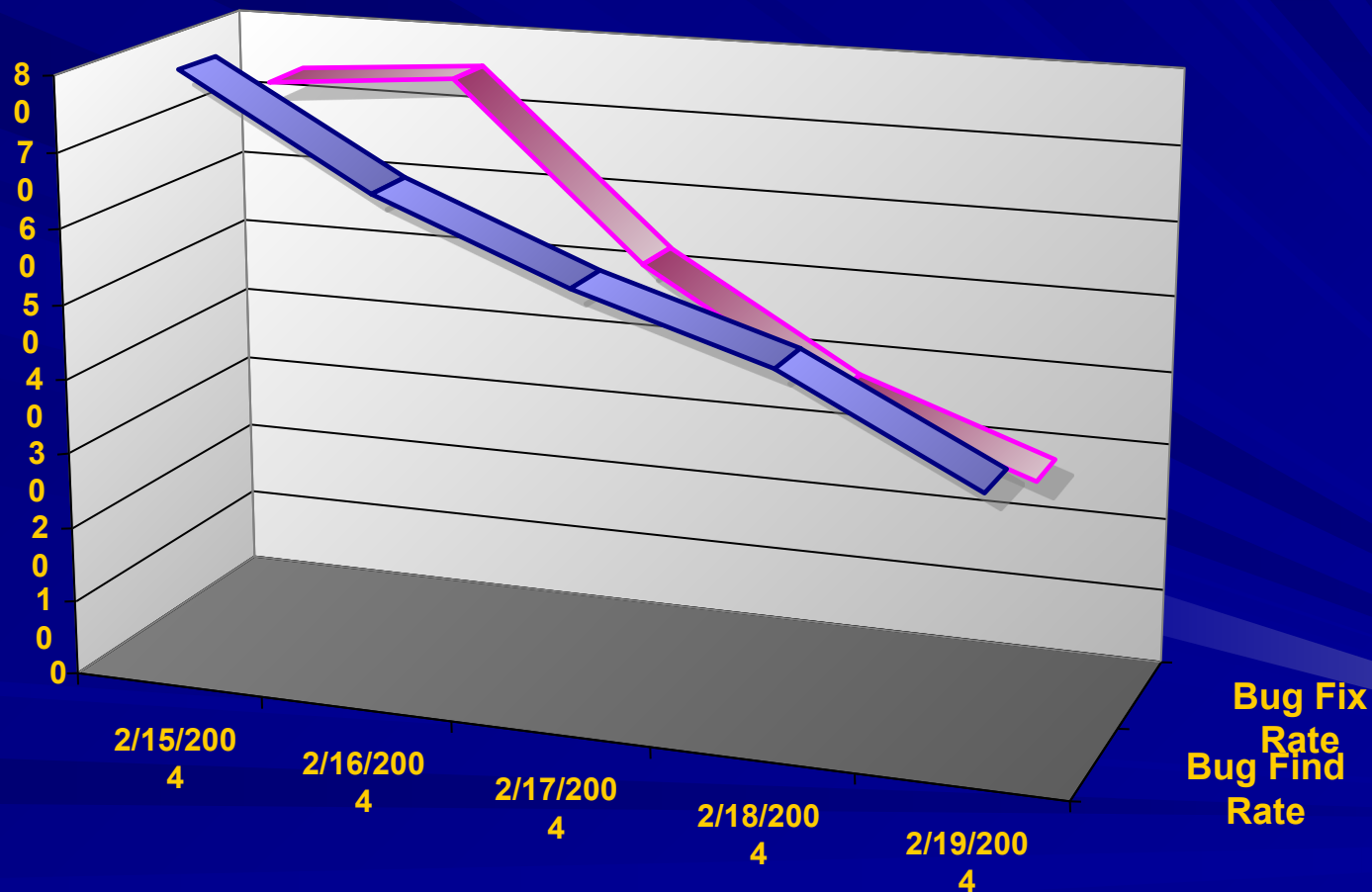
- IsMember
- CreateProjectGroup
- ListProjectGroups
- AddMemberToProjectGroup
- RemoveMemberFromProjectGroup
- ListMembers
- ListMemberOf

# Classification Service



# Класификация не нужна

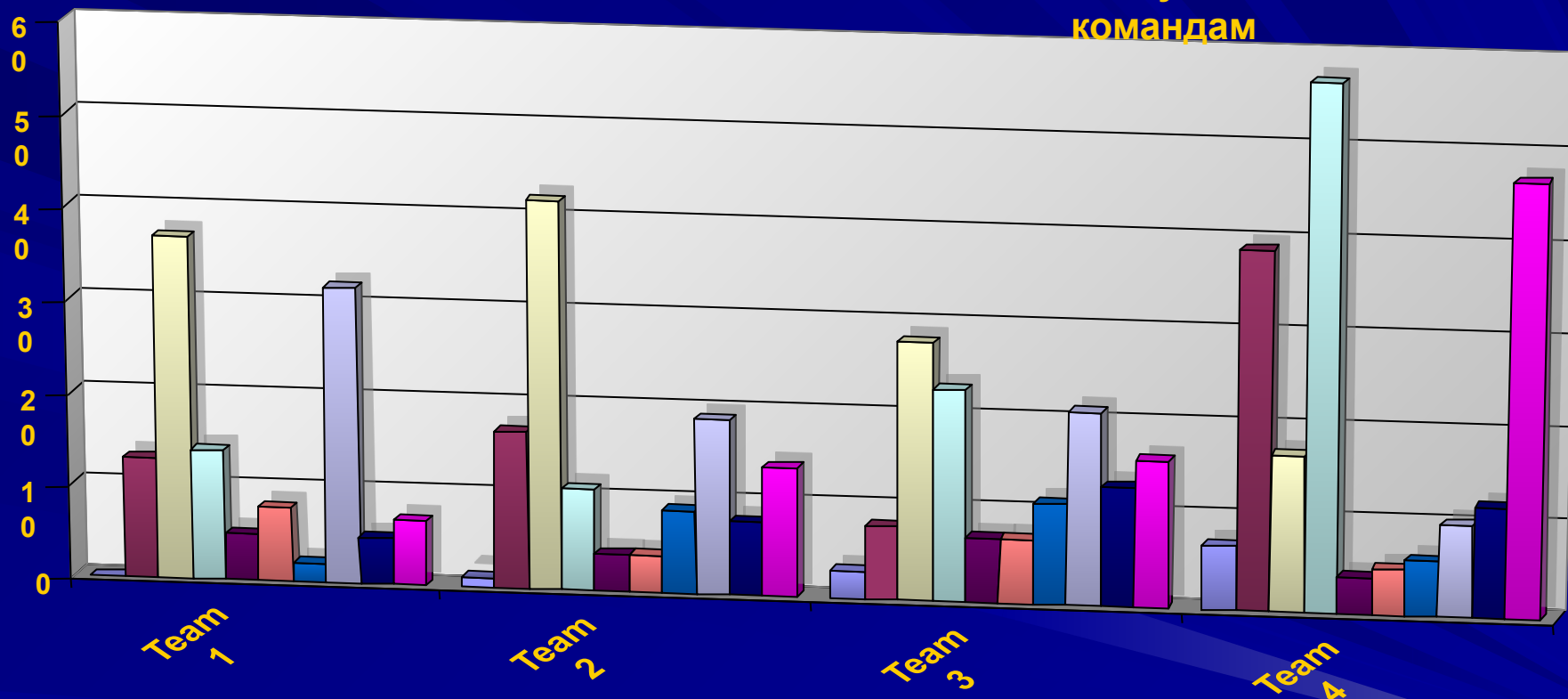
## Bug Trend - Find & Fix



# Классификация нужна

Status by Team

Статус по командам



- |                                       |                             |                              |
|---------------------------------------|-----------------------------|------------------------------|
| ■ Незаконченная Спецификация          | ■ Неоконченные задачи (10s) | ■ Строки кода (1000s)        |
| ■ Новые /измененные строки кода (10s) | ■ Активные баги             | ■ Степень нахождения багов   |
| ■ Степень починки Fix Rate            | ■ Запуски тестов(10s)       | ■ Степень падения тестов (%) |
| ■ Непокрытый тестом код (%)           |                             |                              |

# Классификация нужна

Status by Team



# Обоснование Классификации

- Общая классификация позволяет делать отчеты по всем инструментам.
- Подход:
  - Организация Work item (структуры портфолио проекта portfolio project structure) понятна для менеджеров проекта
  - Пользователи категоризуют source trees/subtrees и другие артефакты (тесты, и т.д.) в структуру work item.
  - Позволяет создавать отчеты, которые не могут следовать из прямых связей между source, work items, и другими интересующими нас артефактами.



# Explicit Классификация и Отчеты

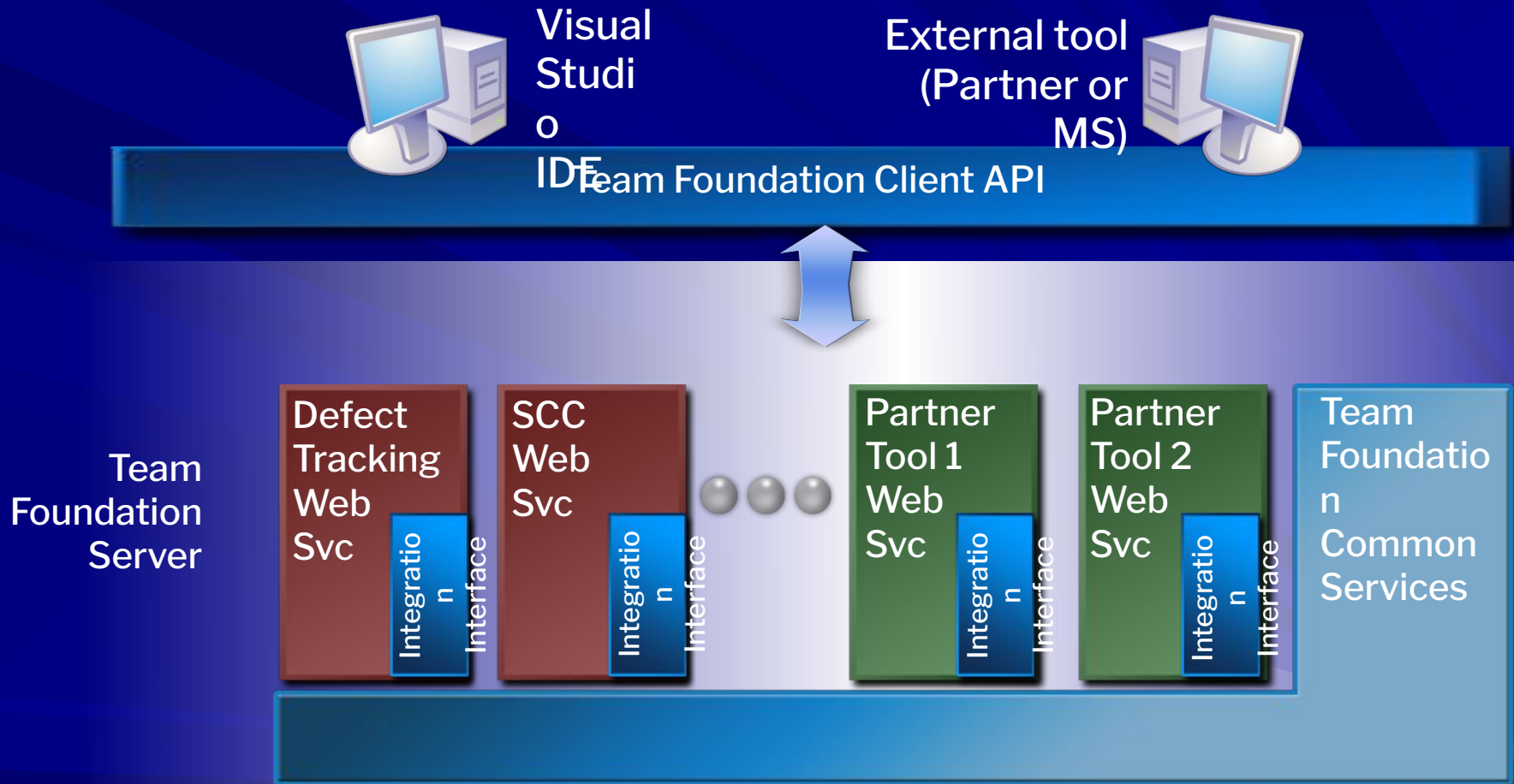
- Некоторые отчеты могут быть написаны без обязательного указания классификации
  - Отчеты из данных полученных из ОДНОГО инструмента.
  - Отчеты из данных из нескольких инструментах, но с преопределенными связями (predefined relationships)
- Некоторые отчеты – не могут....
  - Если связи не predetermined.



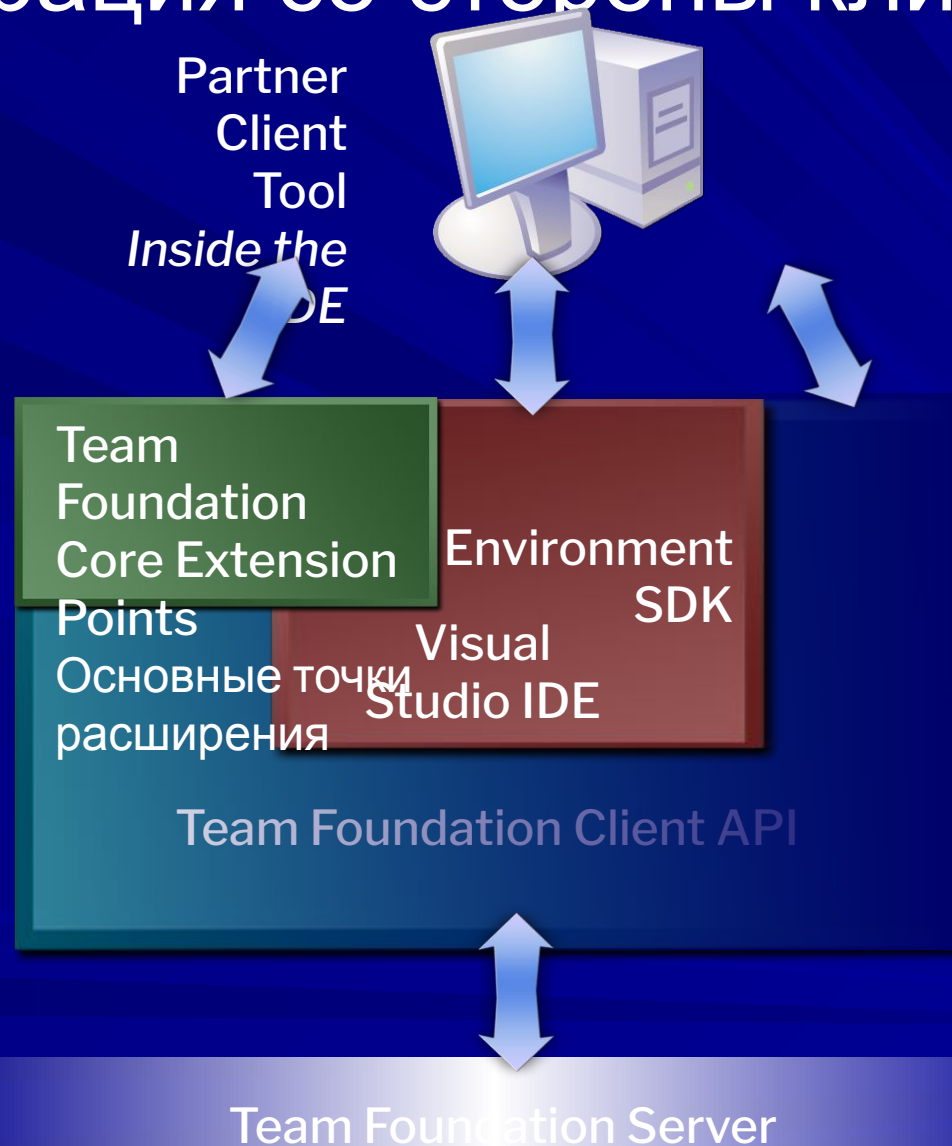
# Обзор API Классификации

- *Артефакты содержат ссылки на классифицирующие ноды (nodes)*
- Контекст проекта - Project Context
  - ListProjectContexts
  - CreateProjectContext
  - DeleteProjectContext
- Структуры
  - GetStructure *(Выдает полную структуру классификаций)*
  - CreateNode
  - ListPermittedChildNodeTypes
  - RenameNode
  - MoveNode
  - DeleteNode

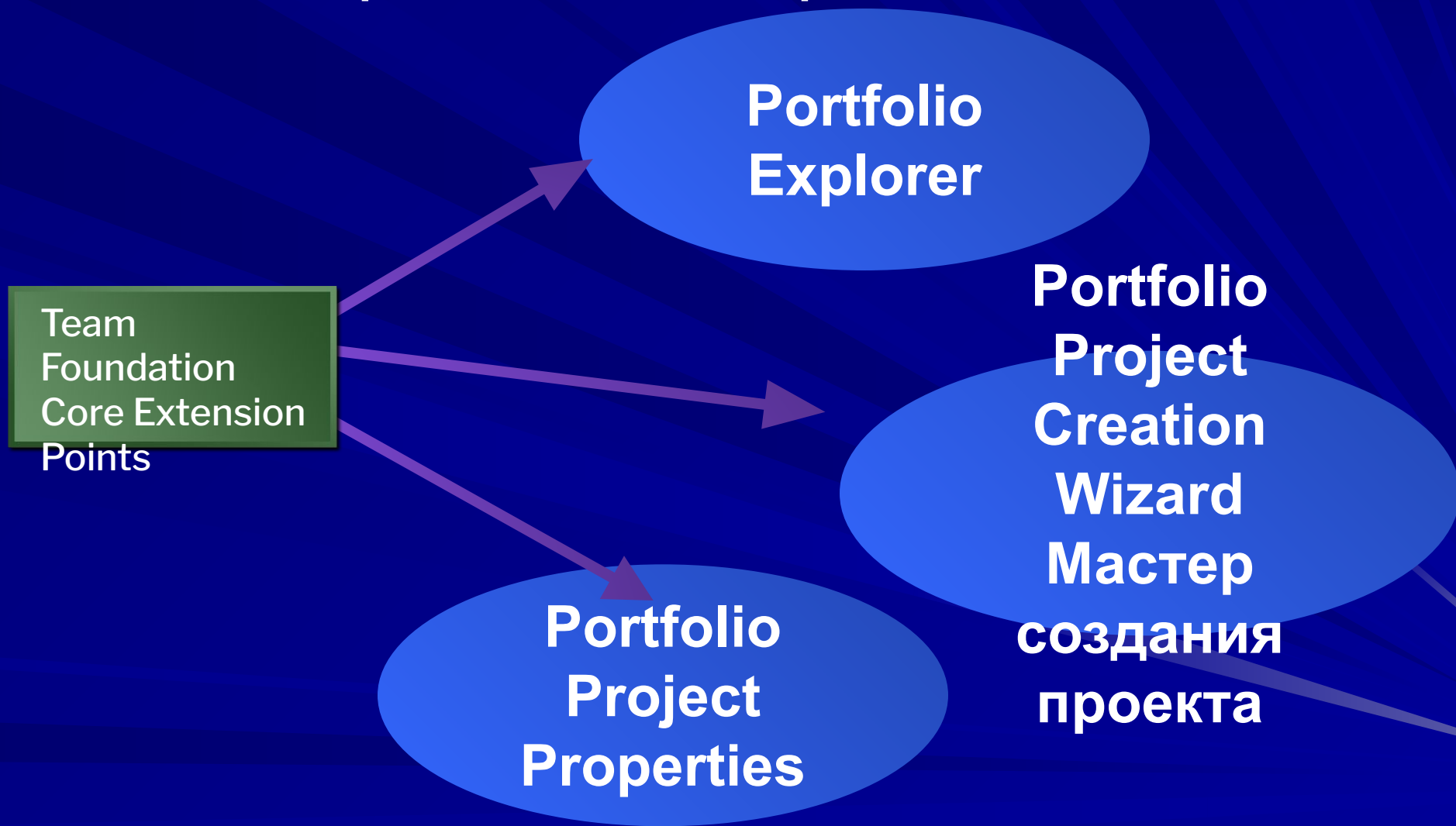
# Интеграция со стороны сервера

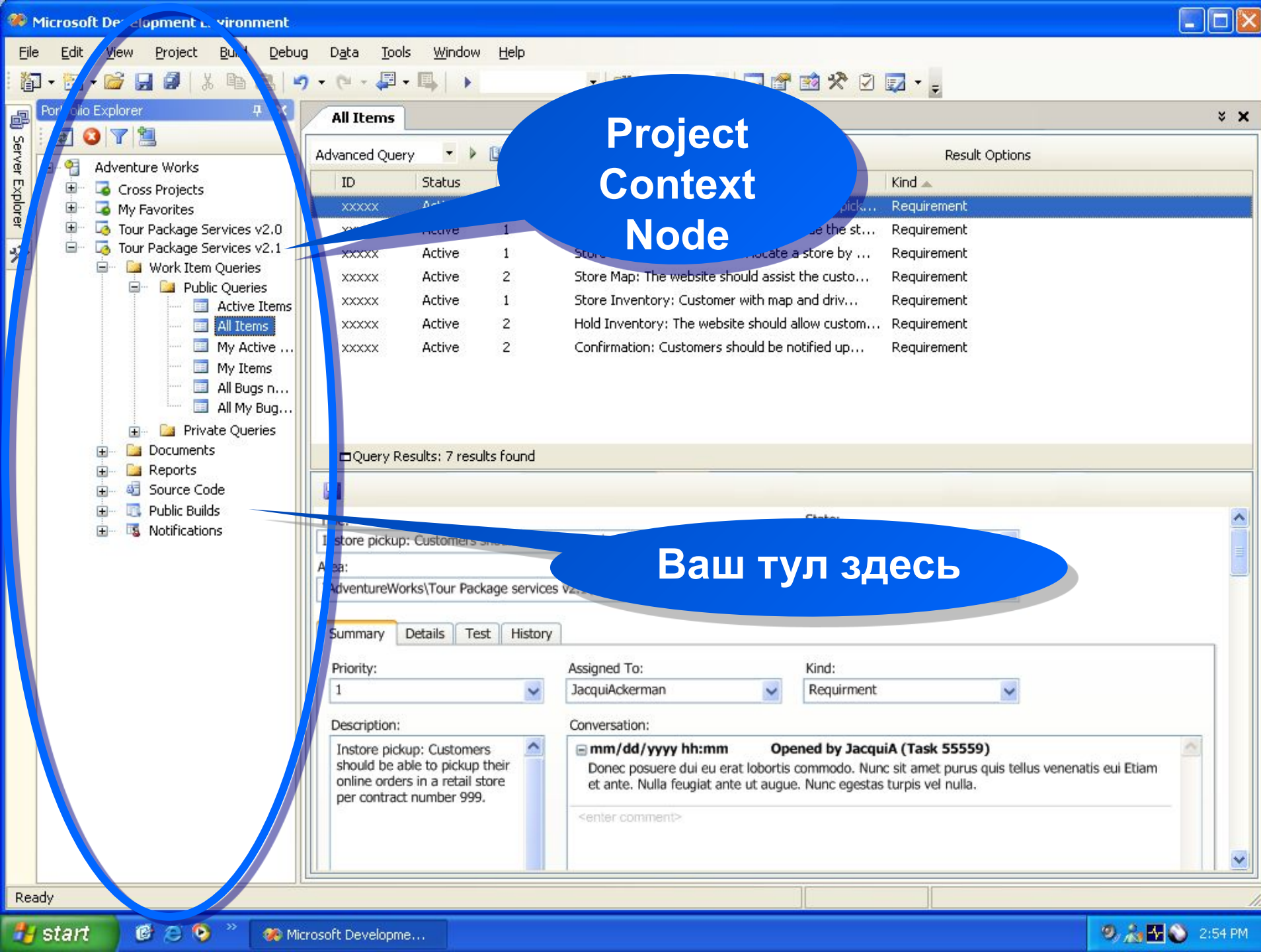


# Интеграция со стороны клиента



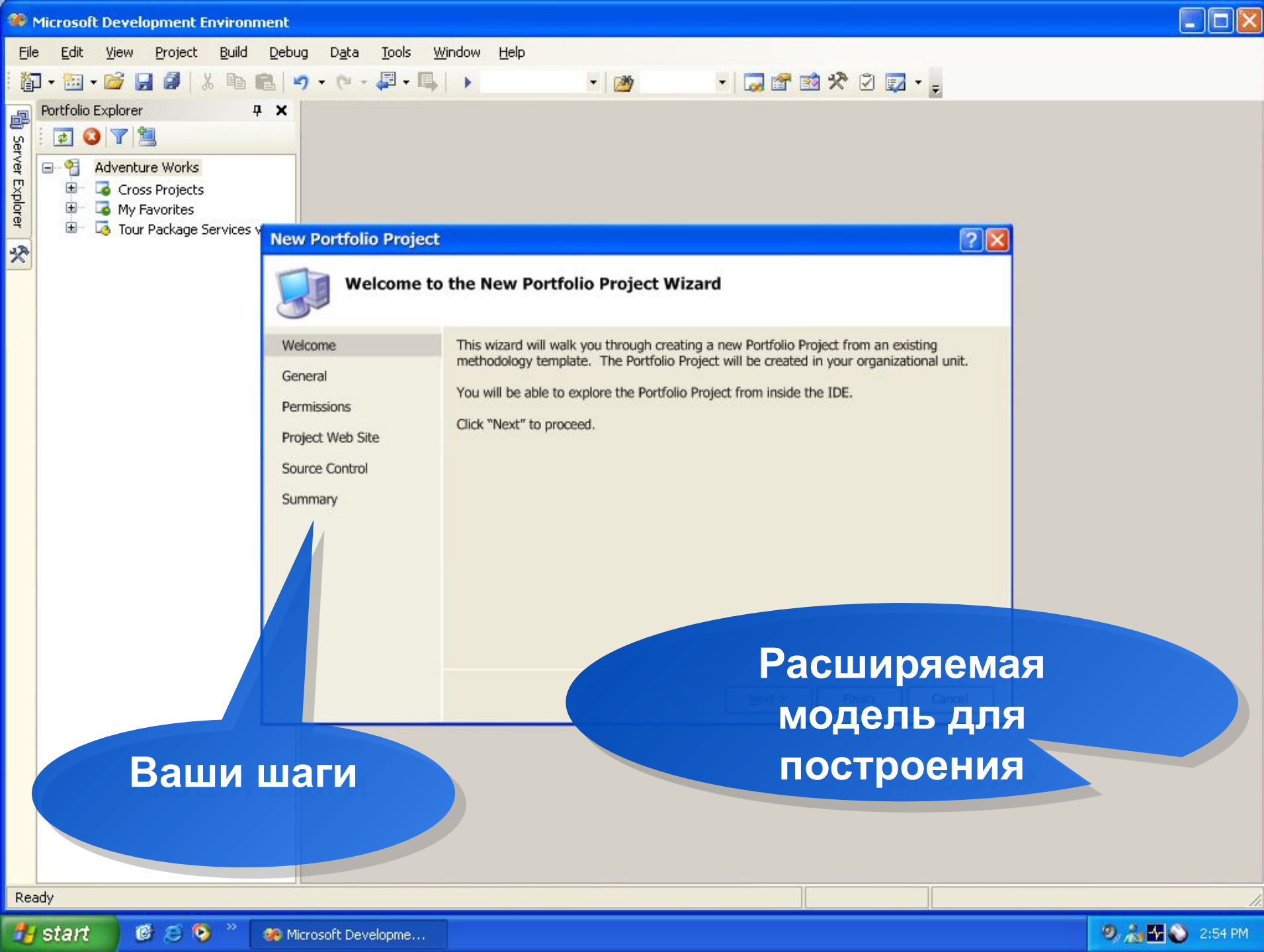
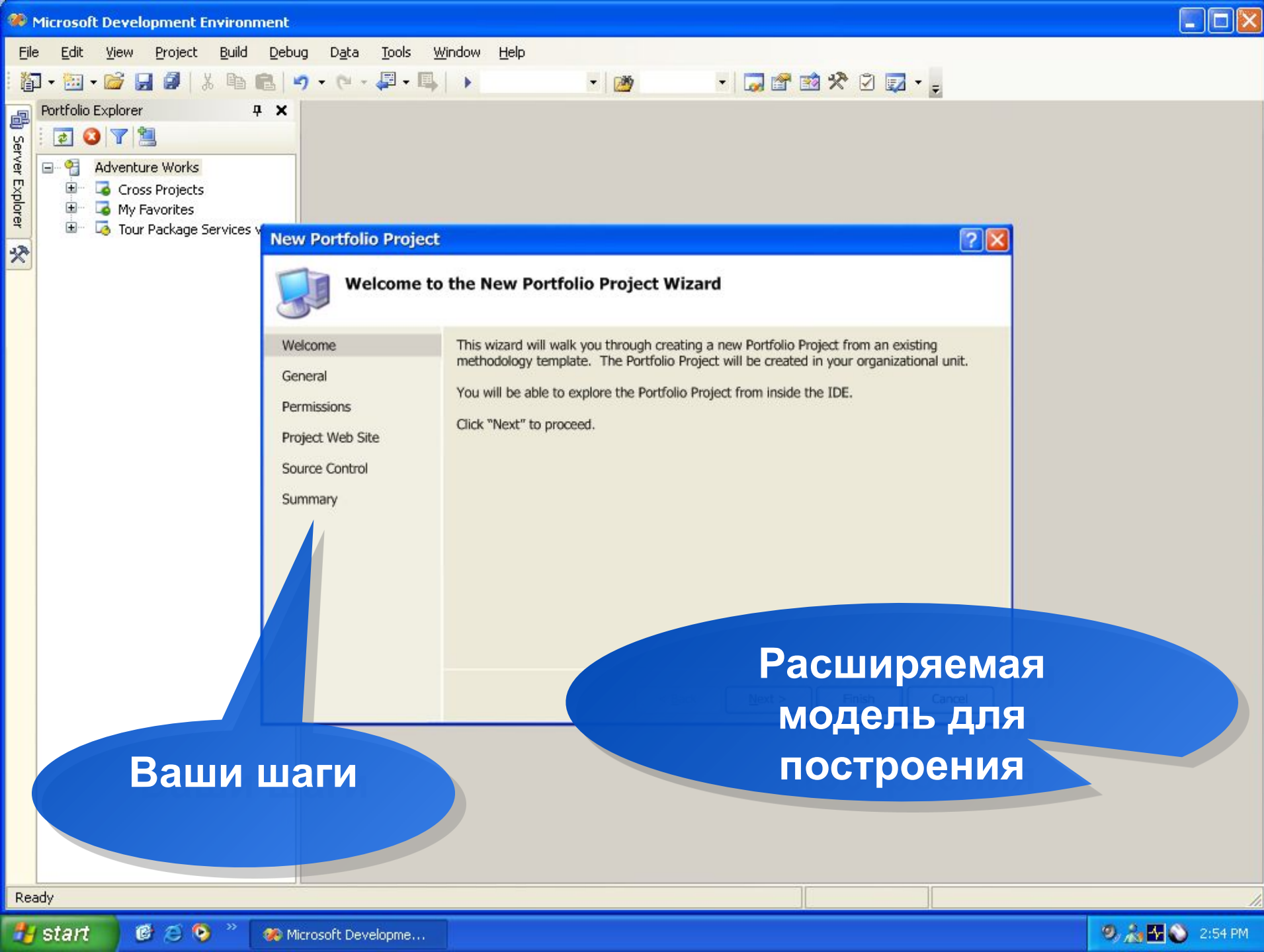
# Интеграция со стороны клиента





Project  
Context  
Node

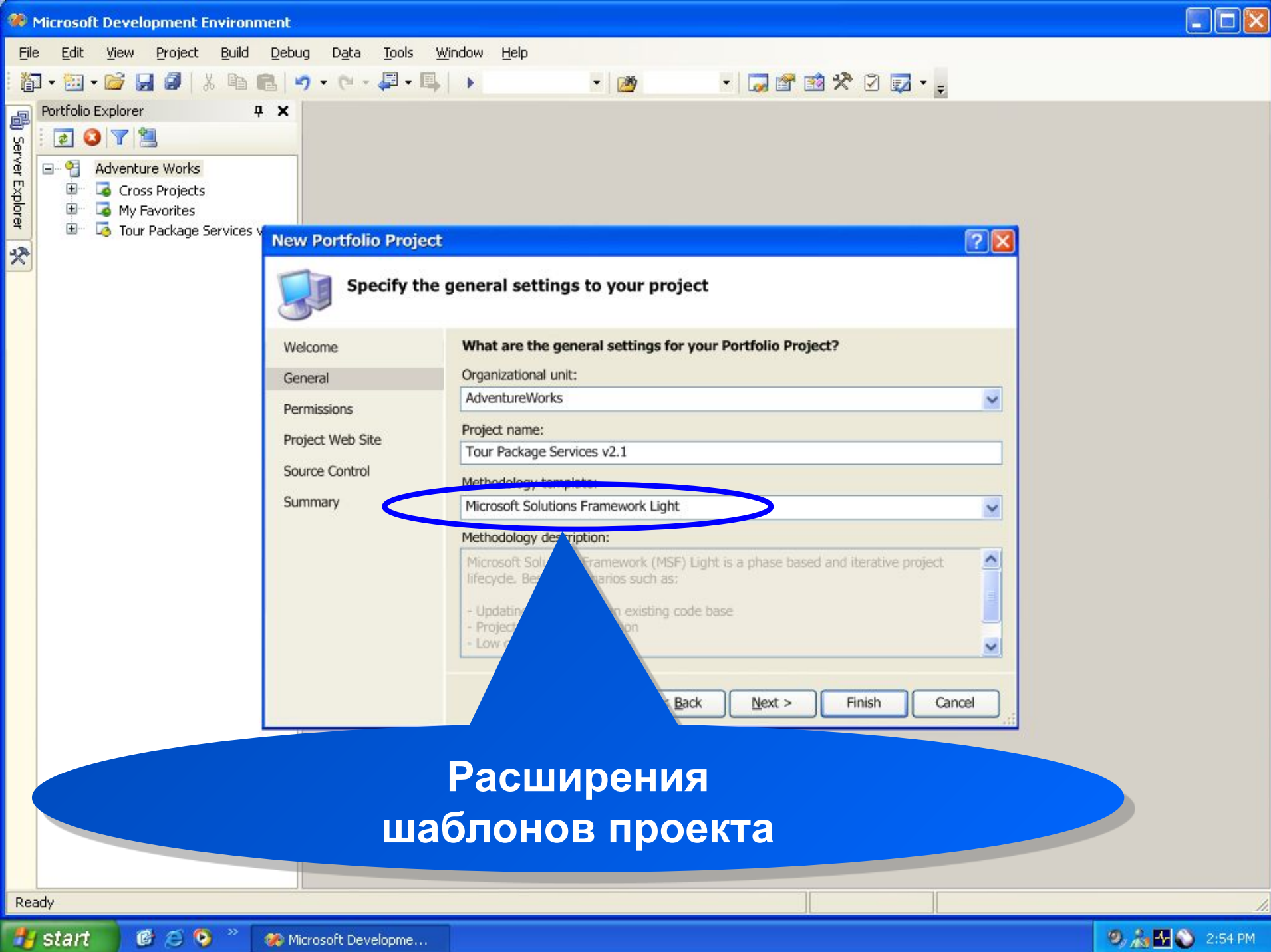
Ваш тул здесь



Ваши шаги

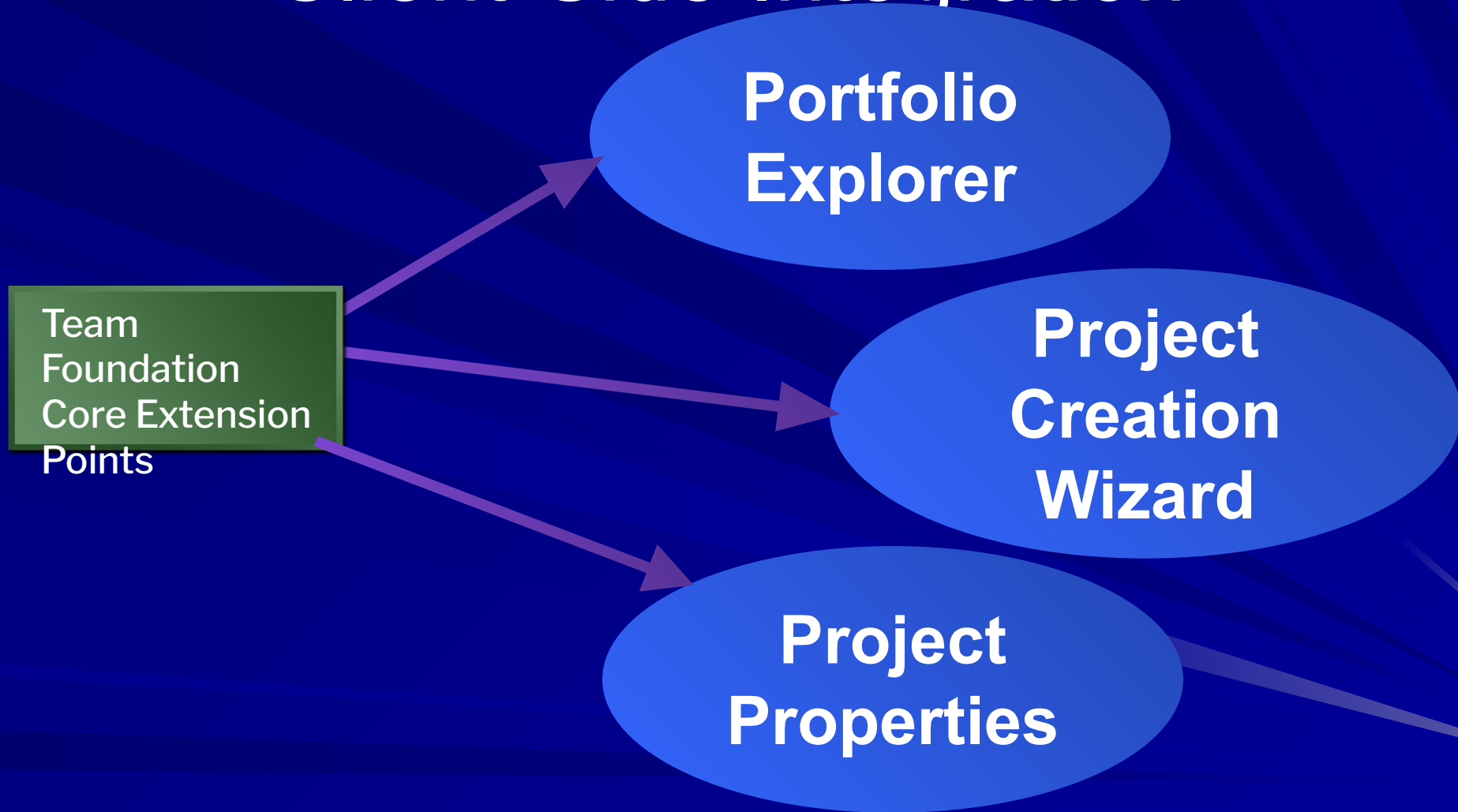
Расширяемая  
модель для  
построения





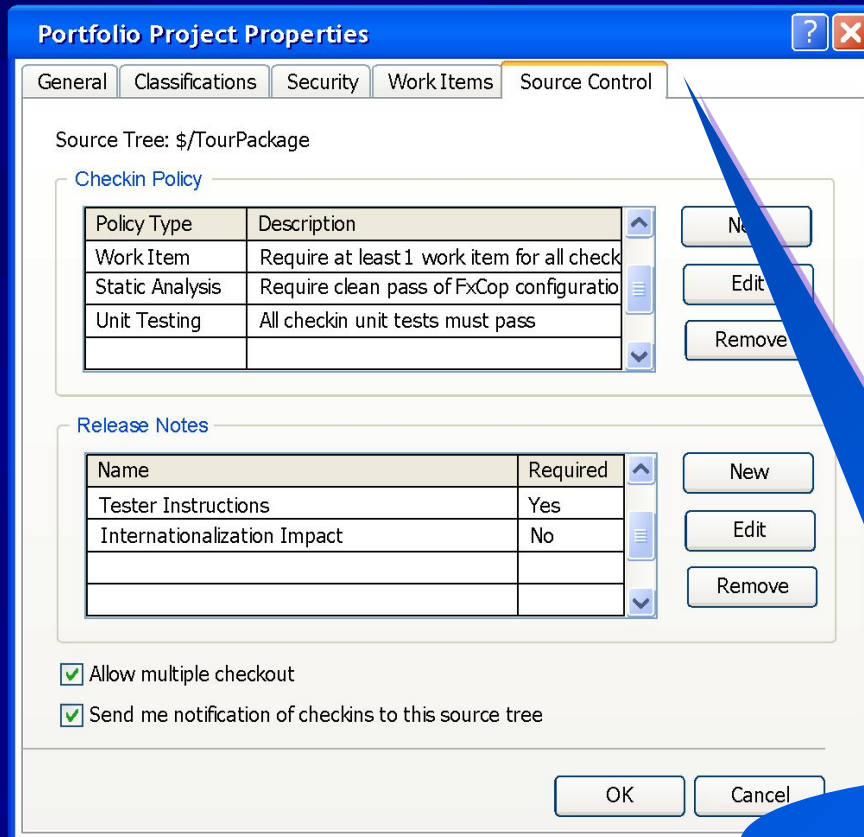
Расширения  
шаблонов проекта

# Client-Side Integration





# Portfolio Project Properties



Portfolio Project Properties

General Classifications Security Work Items **Source Control**

Source Tree: \$/TourPackage

Checkin Policy

Policy Type	Description
Work Item	Require at least 1 work item for all check
Static Analysis	Require clean pass of FxCop configuration
Unit Testing	All checkin unit tests must pass

New Edit Remove

Release Notes

Name	Required
Tester Instructions	Yes
Internationalization Impact	No

New Edit Remove

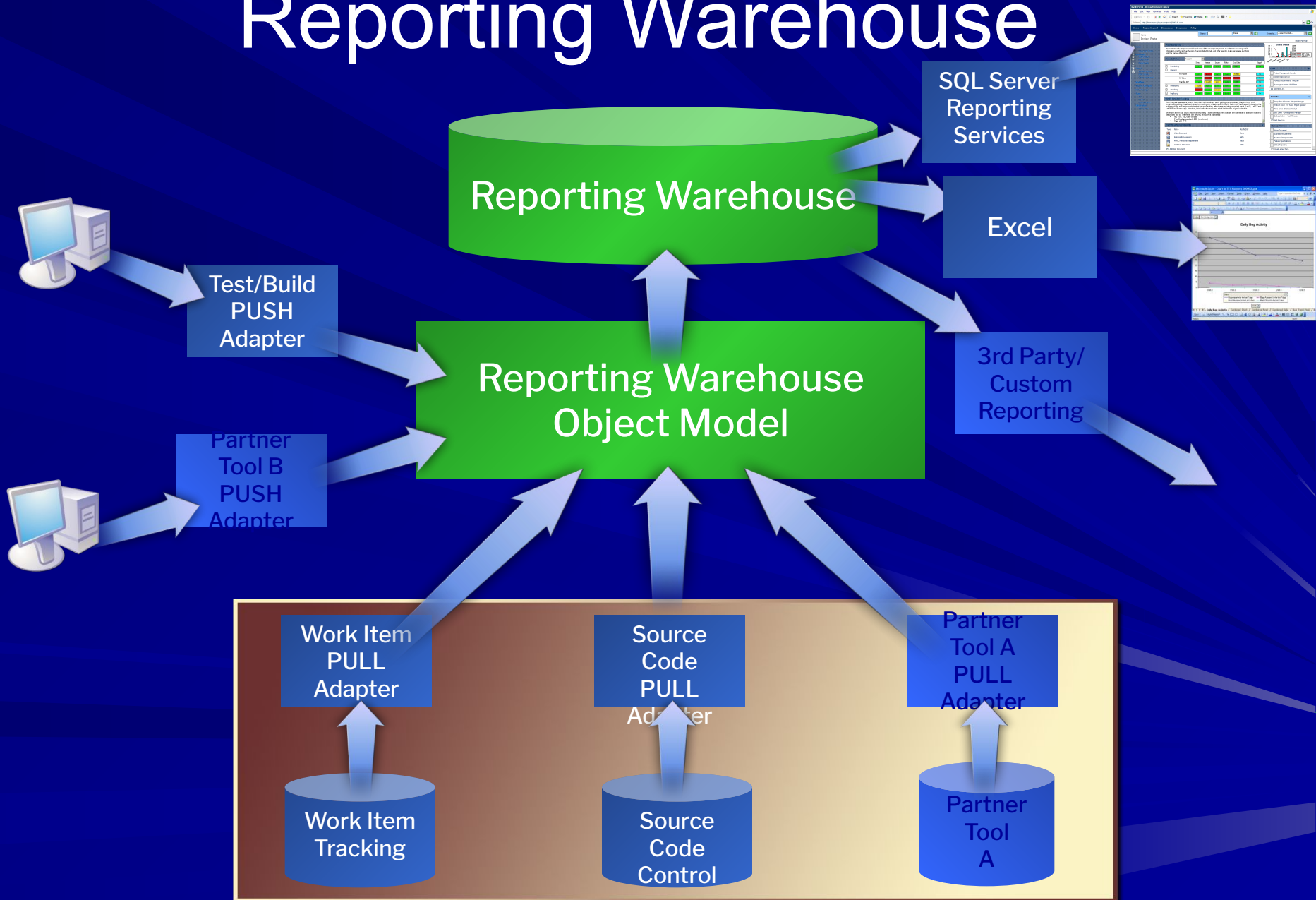
☒ Allow multiple checkout

☒ Send me notification of checkins to this source tree

OK Cancel

Новый таб

# Reporting Warehouse

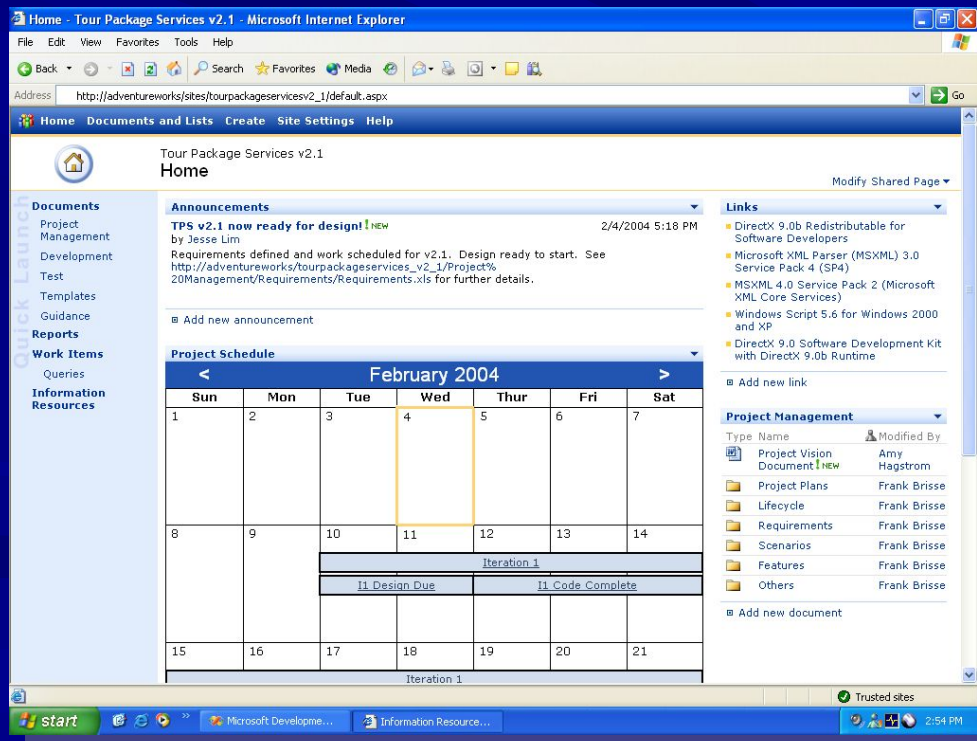


# Reporting Warehouse

- Collection point for reporting/analysis
- Storage for trend analysis
- Extensible
  - Adapter model
  - Add dimensions/measures
- *Pull* model from data source
- *Push* model directly to warehouse
- Uses SQL Server Yukon
  - *RDB for star schema*
  - *Analysis Services for OLAP cubes*

# Project Portal

- WSS Site на проект
- Добавление своих web parts в галерею
- Расширение шаблонов в TFS





## Tour Package Services v2.1 Home

Modify Shared Page ▾

- Documents**
- Project Management
  - Development
  - Test
  - Templates
  - Guidance
- Reports**
- Work Items**
- Queries
- Information Resources**

### Announcements ▾

**TPS v2.1 now ready for design!** !NEW 2/4/2004 5:18 PM  
by Jesse Lim  
Requirements defined and work scheduled for v2.1. Design ready to start. See [http://adventureworks/tourpackageservices\\_v2\\_1/Project%20Management/Requirements/Requirements.xls](http://adventureworks/tourpackageservices_v2_1/Project%20Management/Requirements/Requirements.xls) for further details.

☐ Add new announcement

### Project Schedule ▾

< February 2004 >						
Sun	Mon	Tue	Wed	Thur	Fri	Sat
1	2	3	4	5	6	7
8	9	10	11	12	13	14
Iteration 1						
I1 Design Due			I1 Code Complete			
15	16	17	18	19	20	21
Iteration 1						

### Links ▾

- DirectX 9.0b Redistributable for Software Developers
- Microsoft XML Parser (MSXML) 3.0 Service Pack 4 (SP4)
- MSXML 4.0 Service Pack 2 (Microsoft XML Core Services)
- Windows Script 5.6 for Windows 2000 and XP
- DirectX 9.0 Software Development Kit with DirectX 9.0b Runtime

☐ Add new link

### Project Management ▾

Type	Name	Modified By
	Project Vision Document <span style="color: green;">!NEW</span>	Amy Hagstrom
	Project Plans	Frank Brisse
	Lifecycle	Frank Brisse
	Requirements	Frank Brisse
	Scenarios	Frank Brisse
	Features	Frank Brisse
	Others	Frank Brisse

☐ Add new document



# Примеры из VS 2005 SDK

- Версионный контроль
  - Политика checkin
  - Основы ВК
  - Запрещенные checkin'ы
- Интеграция с Team Build
  - Подписки на TFS
- Основные службы TFS
  - Коллекции
- Work Item tracking
  - Work items
  - Work Items Controls
- Team Explorer
  - Team Explorer плагин
- Team Test
  - Расширения для Team Test – Team Test Extensibility

# Architecting for TFS

- On the server
  - Register your tool characteristics at install time
  - “Artifact-ize” for data integration
    - Implement TFS-enablement APIs/Web Services
    - Add an ASP page for web-based artifact display
  - Extend your authorization to include TFS user groups
  - Raise/respond to TFS Notifications
  - Use (and extend) the warehouse and reports
- On the client
  - Use IDE extensibility points
  - Provide web parts to expose your data
  - Add your own portfolio project and SharePoint templates



The Microsoft logo is centered on a dark blue background. The background features a series of lighter blue diagonal lines that create a sense of motion or depth, radiating from the right side. The logo itself is in a bold, italicized, sans-serif font, with a registered trademark symbol (®) at the end.

# **Microsoft®**

© 2003 Microsoft Corporation. All rights reserved.

This presentation is for informational purposes only. Microsoft makes no warranties, express or implied, in this summary.